-mix

MCR-74-98
NAS5-24067

(NASA-CR-139021) SPACE TRAJECTORY ERROR ANALYSIS PROGRAM (STEAP) FOR HALO ORBIT MISSIONS. VOLUME 2: PROGRAMMER'S MANUAL Study (Martin Marietta Aerospace, Denver, Colo.) ~~258~~ 257 p HC CL 22C G3/30 N74-26308 Unclas 41091

SPACE TRAJECTORY ERROR ANALYSIS PROGRAM (STEAP) FOR HALO ORBIT MISSIONS

VOLUME 2: PROGRAMMER'S MANUAL

D. V. Byrnes
P. C. Carney
J. W. Underwood
E. D. Vogt

Planetary Systems Mission
Analysis and Operations

MARTIN MARIETTA AEROSPACE
P. O. Box 179
Denver, Colorado 80201

May, 1974

Prepared for:
GODDARD SPACE FLIGHT CENTER
Greenbelt, Maryland 20771

257

1. Report No.	2. Government Accession No.	3. Recipient's Catalog No.
MCR-74-98		

4. Title and Subtitle	5. Report Date
SPACE TRAJECTORIES ERROR ANALYSIS PROGRAMS (STEAP) FOR HALO ORBIT MISSIONS	May, 1974
	6. Performing Organization Code
7. Author(s)	8. Performing Organization Report No. MCR-74-98
9. Performing Organization Name and Address Planetary Systems Mission Analysis and Operations Section Martin Marietta Aerospace P. O. Box 179 Denver, Colorado 80201	10. Work Unit No. 11. Contract or Grant No. NAS5-24067
12. Sponsoring Agency Name and Address National Aeronautical and Space Administration Goddard Space Flight Center Greenbelt, Maryland 20771 Technical Monitor: Robert W. Farquhar	13. Type of Report and Period Covered Study Report November, 1973 thru May, 1974 14. Sponsoring Agency Code

15. Supplementary Notes

16. Abstract

The six month effort was responsible for the development, test, conversion, and documentation of computer software for the mission analysis of missions to halo orbits about libration points in the Earth-Sun system. The software consisting of two programs called NOMNAL and ERRAN is part of the Space Trajectories Error Analysis Programs (STEAP) developed by MMC.

The program NOMNAL targets a transfer trajectory from Earth on a given launch date to a specified halo orbit on a required arrival date. Either impulsive or finite thrust insertion maneuvers into halo orbit are permitted by the program. The transfer trajectory is consistent with a realistic launch profile input by the user.

The second program ERRAN conducts error analyses of the targeted transfer trajectory. Measurements including range, doppler, star-planet angles, and apparent planet diameter are processed in a Kalman-Schmidt filter to determine the trajectory knowledge uncertainty. Execution errors at injection, midcourse correction and orbit insertion maneuvers are analyzed along with the navigation uncertainty to determine trajectory control uncertainties and fuel-sizing requirements. The program is also capable of generalized covariance analyses.

The final report consists of two volumes: an Analytic and User's Manual and a Programmer's Manual.

17. Key Words	18. Distribution Statement
1. Halo Orbit 2. Libration Points 3. Mission Analysis 4. Navigation and Guidance 5. Estimation Theory	

19. Security Classif. (of this report)	20. Security Classif. (of this page)	21. No. of Pages	22. Price
UNCLASSIFIED	UNCLASSIFIED		

PREFACE

The objective of this contract (NAS5-24067) is the development of computer software for the preflight mission analysis of missions to earth-sun libration points. This software, designated STEAP-L, extends the capability of the Space Trajectories Error Analysis Programs (STEAP) developed under contracts NAS1-9745, NAS5-11795, and NAS5-11873 and begins the integration of STEAP with the Goddard Trajectory Determination System (GTDS).

The software produced consists of two related programs, both of which use the GTDS Cowell propagator for the computation of the trajectory and state transition matrices. The first program, NOMNAL, is responsible for the generation of the nominal trajectory from launch at earth to insertion into halo orbit about the desired libration point. NOMNAL uses a Newton-Raphson iteration (moving backward in time from the insertion maneuver) to perform the targeting of both impulsive and finite burn insertions into halo orbit. A user-controlled launch profile allows the transfer to be tied to a realistic launch and injection. NOMNAL stores the targeted trajectory and state transition matrices on a file for later analysis by the second program ERRAN.

The program ERRAN performs generalized linear error analyses along specific targeted trajectories. Knowledge and control covariances are propagated along the trajectory through a series of measurements and guidance events in a totally integrated fashion. The knowledge covariance is processed through measurements using a Kalman-Schmidt recursive filter with arbitrary solve-for/consider/ignore state augmentation. Probabilistic midcourse corrections are computed using an exact analytic formulation. ERRAN obtains the trajectory and state transition matrices from a file generated by NOMNAL for program efficiency.

A major conclusion of this effort is that the complementary features of the GTDS and STEAP systems may be effectively combined to yield a significantly improved system. Thus the Cowell file generator/reader capability of the GTDS has been combined with the generalized covariance analysis of STEAP to yield a more efficient, extended error analysis capability than either system had previously. Other conclusions reflect the efficacy of the backward targeting algorithm developed for the libration mission targeting and the analytic formulation implemented for the midcourse correction sizing.

The general recommendations for future effort identified during this study are two-fold. Because of the success of this preliminary integration of the GTDS and STEAP systems it is recommended that this effort be continued and enlarged. In the specific area of libration point mission analysis, it is recommended that more detailed models (e.g., pulsing thrust insertion into halo orbit) be developed and continued studies be made of critical problems (e.g. station-keeping error analysis) for these peculiar missions which are neither interplanetary, lunar, nor earth-orbiting.

TABLE OF CONTENTS

		PAGE
1.	INTRODUCTION	1-1
	1.1 Development of STEAP	1-1
	1.2 Libration Point Mission Application	1-2
	1.3 Summary of NOMNAL	1-4
	1.4 Summary of ERRAN	1-5
2.	NOMNAL PROGRAM STRUCTURE	2-1
	2.1 NOMNAL Program Description	2-1
	2.2 NOMNAL Subroutine Hierarchy	2-1
	2.3 NOMNAL Common Variables	2-1
	2.3.1 Common Blocks in Alphabetical Order	2-2
	2.3.2 Common Variables in Alphabetical Order	2-15
3.	ERRAN PROGRAM STRUCTURE	
	3.1 ERRAN Program Description	
	3.2 ERRAN Subroutine Hierarchy	
	3.3 ERRAN Common Variables	
	3.3.1 Common Blocks in Alphabetical Order	
	3.3.2 Common Variables in Alphabetical Order	
4.	INDIVIDUAL SUBROUTINE DOCUMENTATION	
5.	REFERENCES	

LIST OF ILLUSTRATIONS

Figure	Caption	Page
1.1	Earth-Sun Libration Pointes	1-3
1.2	Details of L_1 and L_2 Libration Points	1-3
1.3	Typical Halo Orbit as Seen from Earth	1-4
2.1	NOMNAL Subroutine Hierarchy	2-2
3.1	ERRAN Subroutine Hierarchy	3-2

LIST OF TABLES

Table	Caption	Page
2.1	NOMNAL Common Blocks in Alphabetical Order	2-3
2.2	NOMNAL Common Variables in Alphabetical Order	2-15
3.1	ERRAN Common Blocks in Alphabetical Order	3-7
3.2	ERRAN Common Variables in Alphabetical Order	
4.1	Alphabetical Listing of Subroutines	4-2
4.2	Purposes of Subroutines	4-3

NOMENCLATURE

A. Arabic Symbols

Symbol	Definition
a	Semi-major axis of conic
C_{xx_s}	Correlation between position/velocity state and solve-for parameters
C_{xu}	Correlation between position/velocity state and dynamic consider parameters
C_{xv}	Correlation between position/velocity state and measurement consider parameters
$C_{x_s u}$	Correlation between solve-for parameters and dynamic consider parameters
$C_{x_s v}$	Correlation between solve-for parameters and measurement consider parameters
e	Eccentricity of conic
E	Eccentric anomaly
f	True anomaly on conic
G	Observation matrix relating observables to dynamic consider parameter state
H	Observation matrix relating observables to position/velocity state
i	Inclination of conic (reference body equatorial)
J	Measurement residual covariance matrix
K	Kalman gain constant for position/velocity state
L	Observation matrix relating observables to measurement consider parameter state
	Mean longitude
M	Observation matrix relating observables to solve-for parameter state
	Mean anomaly
n_1	Dimension of solve-for parameter state
n_2	Dimension of dynamic consider parameter state
n_3	Dimension of measurement consider parameter state
p	Semilatus rectum of conic Probability density function
P	Position/velocity covariance matrix
$\hat{P}$	Unit vector to periapsis of conic

Symbol	Definition
P_s	Solve-for parameter covariance matrix
Q	Dynamic noise covariance matrix
$\tilde{Q}$	Execution error matrix
$\hat{Q}$	Unit vector in plane of motion normal to P
r	Radius
r_{CA}	Radius of closest approach
R	Measurement noise covariance matrix
$\underline{R}$	Actual noise covariance matrix
S	Kalman gain constant for solve-for parameters
S_j	Velocity correction covariance matrix
t_{CA}	Time of closest approach to target body
Δt	Time interval
U_o	Dynamic consider parameter covariance matrix
v	Velocity
V_o	Measurement consider parameter covariance matrix
W_j	Target parameter covariance matrix
$\hat{W}$	Unit normal to orbital plane
X	Actual position/velocity state
$\bar{X}$	Targeted nominal position/velocity state

B. Greek Synbols

Γ_j	Guidance matrix
Γ	Flight path angle
δ	Declination of vector
Δv	Velocity increment
ε	Measurement residual Errors in target parameters
η_j	Variation matrix relating position/velocity variations to target conditions
θ_{xx_s}	State transition matrix partition associated with solve-for parameters
θ_{xu}	State transition matrix partition associated with dynamic consider parameters

Symbol	Definition
θ	Longitude or right ascension
Λ_j	Projection of target condition covariance matrix W_j into the impact plane
μ	Gravitational constant of body
$\vec{\mu}$	Biased aimpoint
ν	Sampled measurement noise True anomaly
ρ	Magnitude of midcourse correction Correlation coefficient
σ	Standard deviation
Σ	Launch azimuth
$\vec{\tau}$	Target parameters
Φ	Targeting matrix State transition matrix for position/velocity state Latitude
χ	Sensitivity matrix
ψ_j	Matrix relating guidance corrections to target condition deviations
Ω	Longitude of ascending node
ω	Argument of periapsis
$\tilde{\omega}$	Longitude of periapsis

C. Subscripts

C	Control variable (P_C)
CA	Closest approach (r_{CA})
f	Final variable (t_f)
i	Initial variable (t_i)
j	Index of current guidance event (P_j)
k	Index of current measurement (P_k)
K	Knowledge variable (P_K)
s	Solve-for parameter (x_s)

D. Superscripts

A	Augmented variable (Φ^A)
T	Matrix transpose (Φ^T)

Symbol	Definition
-1	Matrix inverse (Φ^{-1})
-	Variable immediately before instant (P_k^- or v^-)
+	Variable immediately after instant P_k^+ or v^+)

E. Abbreviations

AU	Astronomical unit
CA	Closest approach to reference body
ERRAN	Error analysis program
FTA	Fixed time of arrival (guidance policy)
GHA	Greenwich hour angle
GSFC	Goddard Space Flight Center
GTDS	Goddard Trajectory Determination System
J.D.	Julian date (referenced either 0^{yr} or 1900^{yr})
km	Kilometers
M/C	Midcourse correction
NOMNAL	Nominal trajectory generation program
S/C	Spacecraft
SF/C	Solve-for/consider
STM	State transition matrix
STEAP	Space Trajectories Error Analysis Programs
VTA	Variable Time of Arrival (Guidance Policy)

1. INTRODUCTION

This Programmer's Manual summarizes the structure and coding of the STEAP-L programs NOMNAL and ERRAN and the subroutines they include. The discussions are intended to provide the reader with sufficient information to effectively modify or extend those programs. An accompanying volume, the Analytic and User's Manual, summarizes the mathematical assumptions and analysis of the programs and details the actual usage (input and output requirements) of those programs.

This volume is divided into three major parts. This introductory chapter discusses the general development of the STEAP library of programs, describes the libration point missions toward which the current effort is directed, and summarizes the capability of the two programs developed for this application, NOMNAL and ERRAN. The second and third chapters summarize the overall programming and storage of the NOMNAL and ERRAN programs respectively. The fourth chapter forming the bulk of this volume provides documentation of each subroutine of STEAP-L in alphabetical order.

1.1 Development of STEAP

STEAP is an acronym for Space Trajectory Error Analysis Programs. Rather than a single computer program, STEAP is a library of related programs for the analysis of the navigation and guidance characteristics of space missions. These programs have been developed, modified, and extended over a number of years by the Martin Marietta Corporation (MMC) under the direction of NASA in a variety of contracts.

There are two primary unifying elements in the development of the STEAP system. The first is in the underlying philosophy of STEAP. STEAP has always been directed toward the performance of a totally-integrated analysis of the navigation and guidance processes of space missions. Thus interaction is continually forced between the tracking uncertainties and the maneuver execution errors to determine the evolving uncertainties in the knowledge and control of the spacecraft trajectory. The second element is in general program structure. The STEAP software has continually been divided into three distinct operational modes responsible for nominal trajectory targeting and generation (NOMNAL), linear error analyses (ERRAN), and single-case or Monte Carlo simulations (SIMUL). The current effort does not address the third of these types of programs.

The mathematical foundation for the STEAP system was initially developed under Contract NAS8-21120 for Marshall Space Flight Center. The first version of STEAP (Contract NAS1-8745) was constructed for general interplanetary ballistic missions for Langley Research Center to support the Viking mission analysis and design. Later development of STEAP was performed for Goddard Space Flight Center (Contracts NAS5-11795 and NAS5-11873) where specific extensions required for Planetary Explorer (later known as Pioneer Venus) and general lunar missions were added in a version called STEAP-II. More recently, programs for the navigation and guidance analysis of low thrust inter-planetary

and near-Earth missions have been developed for Langley Research Center (NAS1-11686) and Marshall Space Flight Center (Contract NAS8-29666). Throughout this time, improvements in the analytical techniques and program structure have been continually identified and incorporated into the STEAP series of programs. (References 1-5).

Under the current contractual effort, versions of NOMNAL and ERRAN appropriate for missions to Earth-Sun libration points have been developed (termed STEAP-L). A very significant feature of this effort is that the Goddard Trajectory Determination System (GTDS) Cowell propagator is being integrated into the STEAP-L programs. The Cowell propagator permits the generation of a file containing trajectory and state transition matrix (computed by integration of the variational equations) data during the NOMNAL run. This data may then be efficiently retrieved in subsequent ERRAN runs, thereby eliminating the costly integration cycle from ERRAN.

A number of new analytical features have been added to STEAP under this contract. An unusual approach has been used in the targeting of the libration point missions. Backward integration is used in computing the successive trajectory iterates and targeting matrices required by the Newton Raphson targeting algorithm. This backward targeting scheme efficiently produces a targeted transfer trajectory that is consistent with realistic launch and injection constraints. The approach is well-suited to cometary or lunar missions as well.

An exact computation of the probabilistic midcourse correction requirements using the recently published technique of Lee-Boain (Reference 6) has been added to ERRAN. This replaces the previous model which employed the Hoffman-Young approximation (Reference 7) and which could lead to significant errors at the higher probability levels. This technique is applicable to lunar or interplanetary trajectories as well as the libration point missions.

A third significant item developed during this effort has been the reformulation of the variable time-of-arrival (VTA) guidance policy for the libration point mission application. The guidance policies available in previous versions of STEAP always assumed that the target state was referenced to a gravitational body such as the moon or a planet. This restriction has now been removed.

The characteristics of the libration point missions necessitating these extensions are described in the summary of the libration point missions given in the next section. The capabilities of the resulting programs NOMNAL and ERRAN are then detailed in the next two sections.

1.2 Libration Point Mission Application

The STEAP-L programs developed under this contract are designed for use primarily for the analysis of missions to the two Earth-Sun libration points near the Earth. These are designated L_1 and L_2 in Figure 1.1, which shows schematically the location of all five classical Lagrangian or libration points.

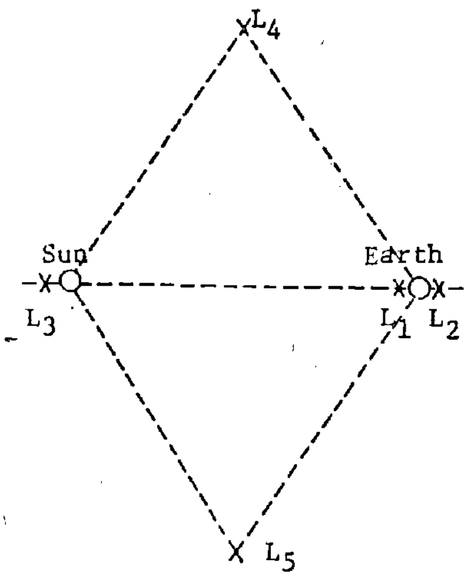


Figure 1.1 Earth-Sun Libration Points

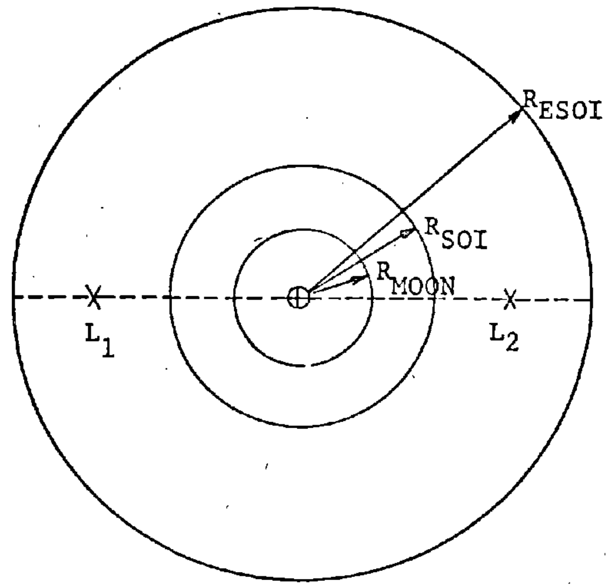


Figure 1.2 Details of L_1 and L_2 Libration Points

Figure 1.2 shows in more detail the location of points L_1 and L_2 with respect to the Earth, with the orbit of the Moon, the classical or Laplacian sphere of influence of the earth, and an enlarged version occasionally used in targeting of swingby missions. The two spheres of influence are defined by

$$R_{SOI} = R_{SE}(M_E/M_S)^{2/5}$$

$$R_{ESOI} = R_{SE}(M_E/M_S)^{1/3}$$

where R_{SE} is the Earth-Sun distance and M_E and M_S are the masses of the Earth and Sun respectively.

Efficient transfers from circular Earth parking orbit to the L_1 and L_2 points have been shown (Reference 8) to fall into at least two major families; those with short (~25 to 50 day) transfer times and those with long (~100 to 135 day) transfer times. The fast transfers require from 341 to about 400 meters/second ΔV to insert into orbit near the libration point, with the minimum ΔV at about 36.4 days. The slow transfers require insertion ΔV of from 272 to about 400 meters/second, with the minimum ΔV at about 116.8 days. These optimum insertion values are based upon the Earth in a circular orbit around the Sun and will vary slightly due to the ellipticity of the orbit of the Earth. The influence of the moon will affect them also. Both of the families discussed above assume a posigrade transfer orbit upon leaving the Earth; corresponding families exist for retrograde departures, but these require higher insertion ΔV at the libration point. For long flight times at least two other families of trajectories exist but have higher ΔV requirements. Even more families exist with longer flight times (~175 days) that have lower ΔV requirements (~200 meters/second) (Reference 8).

The primary feature of the libration points is that they are equilibrium points of the system; i.e., if a spacecraft is placed exactly at a libration point with no motion relative to the system, it will remain at that point relative to the two-body configuration. The collinear points (L_1, L_2, L_3) are unstable while the equilateral triangle points (L_4, L_5) are only quasi-stable. Thus, some form of station-keeping is necessary to maintain the spacecraft in that location. However, the fuel required is still much less than it would be at arbitrary points of the system. Thus, the L_1 and L_2 points offer attractive stations for spacecraft for monitoring solar or solar/earth phenomena (Reference 9). To facilitate communications, the spacecraft would generally be placed in a "halo-orbit" about the libration point so that the sun would not obstruct the view of the spacecraft from earth. A typical halo-orbit in the plane normal to the rotating earth-sun line is illustrated in Figure 1.3.

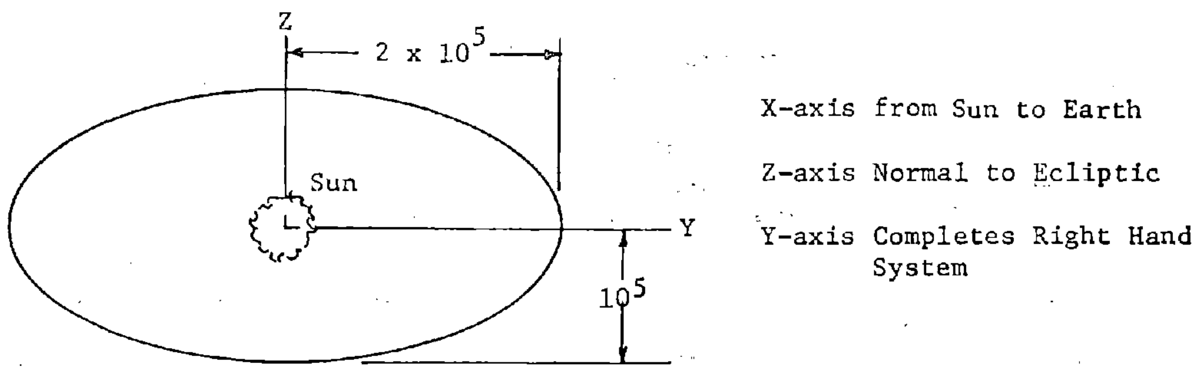


Figure 1.3 Typical Halo-Orbit as Viewed from Earth

The current effort is directed toward the study of the transfer and insertion phases of the libration point mission; the station-keeping while in the halo-orbit was not addressed in this effort. The two programs developed for the analysis of libration point transfers include the nominal trajectory and maneuver targeting program NOMNAL and the navigation and guidance error analysis program ERRAN summarized in the next two sections.

1.3 Summary of NOMNAL

The computer program NOMNAL is responsible for the generation of a nominal trajectory from injection at earth to insertion into a halo orbit about a libration point in the earth-sun system.

NOMNAL uses a specialized version of the GTDS Cowell propagator for the integration of the trajectory equations. The dynamic model used in the reduced Cowell propagator includes the accelerations, on the spacecraft produced by a central body, up to two non-central bodies, and finite thrust engines. The Cowell propagator generates state and control transition matrices by integration of variational equations simultaneously with the equations of motion. These matrices are then used in the targeting of the libration point missions within NOMNAL and in the propagation of covariance matrices and the error analysis of the finite burn insertion maneuver in ERRAN.

NOMNAL has the capability to target transfer trajectories to libration points using both impulsive and finite thrust insertion maneuvers. In either case a backward targeting scheme is employed where conditions at the libration point are iteratively improved to yield trajectories which when propagated backwards in time from the desired arrival point and time to the earth satisfy desired target conditions. The three target conditions at the earth are radius of closest approach, equatorial inclination at closest approach, and time at closest approach. These three conditions are normally selected to be consistent with the desired parking orbit radius, launch site latitude, and desired trip time.

In impulsive targeting the three controls at the libration point are the three components of velocity on the transfer trajectory. In finite thrust targeting the controls are the right ascension and declination of the thrust direction and the duration of the burn; the thrust magnitude, engine specific impulse, and initial spacecraft mass are held constant at the user-supplied values. A Newton-Raphson algorithm is used to iteratively improve the control parameters to determine their required values.

The program includes three options for the determination of the zero iterate values to begin the targeting process: table interrogation, conic approximation, and user-specification. Tables defining targeted velocities have been constructed for transfers to the L_1 and L_2 points with trip times in the vicinity of either optimal transfer (tabulated ΔVs for trip times of from 25 to 50 days and from 102 to 130 days at 1 day intervals). Initial values of velocity may then be interpolated from the data stored in these tables. The second option computes the initial libration point velocity by solving Lambert's theorem for the geocentric conic connecting the libration point radius and the injection radius in the desired time. The third option accepts a user-supplied zero iterate vector computed by the user outside the program.

NOMNAL can adjust the injection time of the transfer to correspond to a realistic launch profile specified by the user. It then adjusts the arrival time by the same amount to hold the trip time at the user-desired value. NOMNAL computes and records such information as the required launch azimuth, coast time, and whether or not a coplanar injection maneuver is required.

1.4 Summary of ERRAN

The error analysis/generalized covariance analysis program ERRAN is a preflight mission analysis tool that is used to determine how selected error sources influence the orbit determination process for libration point missions.

In the error analysis mode, ERRAN provides three primary quantitative results: (1) knowledge covariance matrices, which provide a measure of how well the actual trajectory is known, (2) control covariance matrices, which when propagated forward to the target provide a measure of how well the nominal target conditions will be satisfied by the actual trajectory, and (3) statistical midcourse ΔVs, which provide a measure of the amount of fuel required for a successful mission.

In the generalized covariance analysis mode, ERRAN provides all of the above information plus corresponding "actual" statistical information. The three results discussed in the previous paragraph are all computed on the basis of statistical distributions assumed by the navigation filter to describe the significant error sources. In the generalized covariance analysis mode, "actual" knowledge covariances, control covariances, and statistical midcourse ΔVs are computed on the basis of statistical distributions that actually describe both error sources acknowledged by the navigation filter and the error sources ignored. The primary use of the generalized covariance analysis program is to study the sensitivity of filter performance to off-design conditions.

Up to 15 measurement parameters may be solved-for or considered by the navigation filter employing a Kalman-Schmidt sequential formulation. Parameters not acknowledged in design of the filter may be treated as ignore parameters when ERRAN is run in the generalized covariance analysis mode. Measurement biases include biases in the locations of the three earth-based tracking stations, and biases in all measurements. Available measurement types are range, Doppler, and a simple optical model. Measurement noise for each measurement type is assumed to be constant.

The computational procedure in ERRAN is divided into basic cycle computations and event computations. Basic cycle computations are concerned with the propagation of covariances forward to a measurement time and processing the measurement. Events refer to a set of specialized computations, not directly concerned with measurement processing, that can be scheduled to occur at arbitrary times along the trajectory. State transition matrices interpolated from the file created by NOMNAL are used for all covariance matrix propagation.

The four events available in ERRAN are eigenvector, prediction, guidance, and final insertion into halo orbit. At an eigenvector event the position and velocity partitions of the knowledge covariance matrix are diagonalized to reveal geometric information about the size and orientation of the position and velocity navigation uncertainties. At a prediction event the most recent covariance matrix is propagated forward to some critical trajectory time to determine predicted navigation uncertainties in the absence of further measurements.

The guidance event is the most complex event and yields much useful information for preflight mission analysis. Several types of guidance events are available in ERRAN. At a midcourse guidance event the user can choose from either fixed or variable time of arrival guidance policies (FTA or VTA). Execution error statistics are generated using an impulsive error model defined by a proportionality error, a resolution error, and two pointing angle errors. The execution errors of the insertion maneuver may be modeled as either an impulsive maneuver (defined above) or a finite thrust maneuver (component errors modeled as two pointing errors and a thrust magnitude uncertainty). The target condition covariance matrix both before and after the maneuver is printed out for midcourse and insertion maneuvers.

2. NOMNAL PROGRAM STRUCTURE

2.1 NOMNAL PROGRAM DESCRIPTION

THE STEAP-L/NOMNAL PROGRAM IS A DIGITAL COMPUTER PROGRAM WRITTEN IN FORTRAN IV COMPATIBLE WITH THE IBM 360/370 SYSTEM. THE PROGRAM CONTAINS ABOUT EIGHTY SUBROUTINES OF WHICH ONE-HALF ARE ASSOCIATED WITH THE COWELL PROPAGATOR, ONE THIRD ARE GENERAL UTILITY ROUTINES, AND THE REMAINDER ARE SPECIFIC ROUTINES FOR THE LIBRATION POINT APPLICATION. THE PROGRAM ACCESSES FOUR DATA SETS DURING OPERATION: THE STANDARD FORTRAN INPUT (SYSIN) AND THE PRESENTATION OUTPUT (SYSOUT=A) DATA SETS, THE DIRECT ACCESS SOLAR/LUNAR/PLANETARY EPHEMERIS FILE USED BY THE GTDS (DSN=GTDS.SLP1950.JAN71) AND THE SEQUENTIAL ORBIT FILE ON WHICH THE TRAJECTORY AND STATE TRANSITION MATRIX DATA ARE STORED FOR LATER USE. THE PROGRAM (NON-OVERLAID) REQUIRES 255K OR 245K BYTES STORAGE DEPENDING UPON WHETHER THE ORBIT FILE IS GENERATED OR NOT.

2.2 NOMNAL SUBROUTINE HIERARCHY

FIGURE 2.1 ILLUSTRATES THE GENERAL PROGRAM STRUCTURE. THE MAIN PROGRAM CALLS THREE EXECUTIVE ROUTINES FOR THE THREE MAJOR ACTIVITIES OF THE PROGRAM. HPRELM IS RESPONSIBLE FOR PRELIMINARY DATA MANIPULATION AND THE GENERATION OF THE ZERO ITERATE SOLUTION. HGIDNS IS RESPONSIBLE FOR THE TARGETING OF EITHER THE IMPULSIVE OR FINITE BURN INSERTION TRAJECTORY. HTRJTY PROPAGATES AND STORES THE REQUIRED TRAJECTORIES AND STATE TRANSITION MATRICES USING THE GTDS COWELL PROPAGATOR SPECIALIZED TO THIS APPLICATION. CHAPTER 4 PROVIDES AN INDEX OF ALL ROUTINES USED IN NOMNAL AND GIVES INDIVIDUAL ROUTINE DOCUMENTATION INCLUDING DESCRIPTIONS, ANALYSES, AND FLOWCHARTS.

2.3 NOMNAL COMMON VARIABLES

THE FOLLOWING TWO SUBSECTIONS PROVIDE DEFINITIONS OF THE VARIABLES APPEARING IN COMMON BLOCKS IN THE NOMNAL PROGRAMS. SUBSECTION 2.3.1 LISTS THE COMMON BLOCKS IN ALPHABETICAL ORDER AND GIVES THE SIZE OF THE BLOCKS AND THE DEFINITIONS OF THE VARIABLES APPEARING WITHIN EACH BLOCK. SUBSECTION 2.3.2 LISTS ALL THE COMMON VARIABLES IN ALPHABETICAL ORDER AND DEFINES THE BLOCK TO WHICH THEY BELONG AND THEIR DEFINITION.

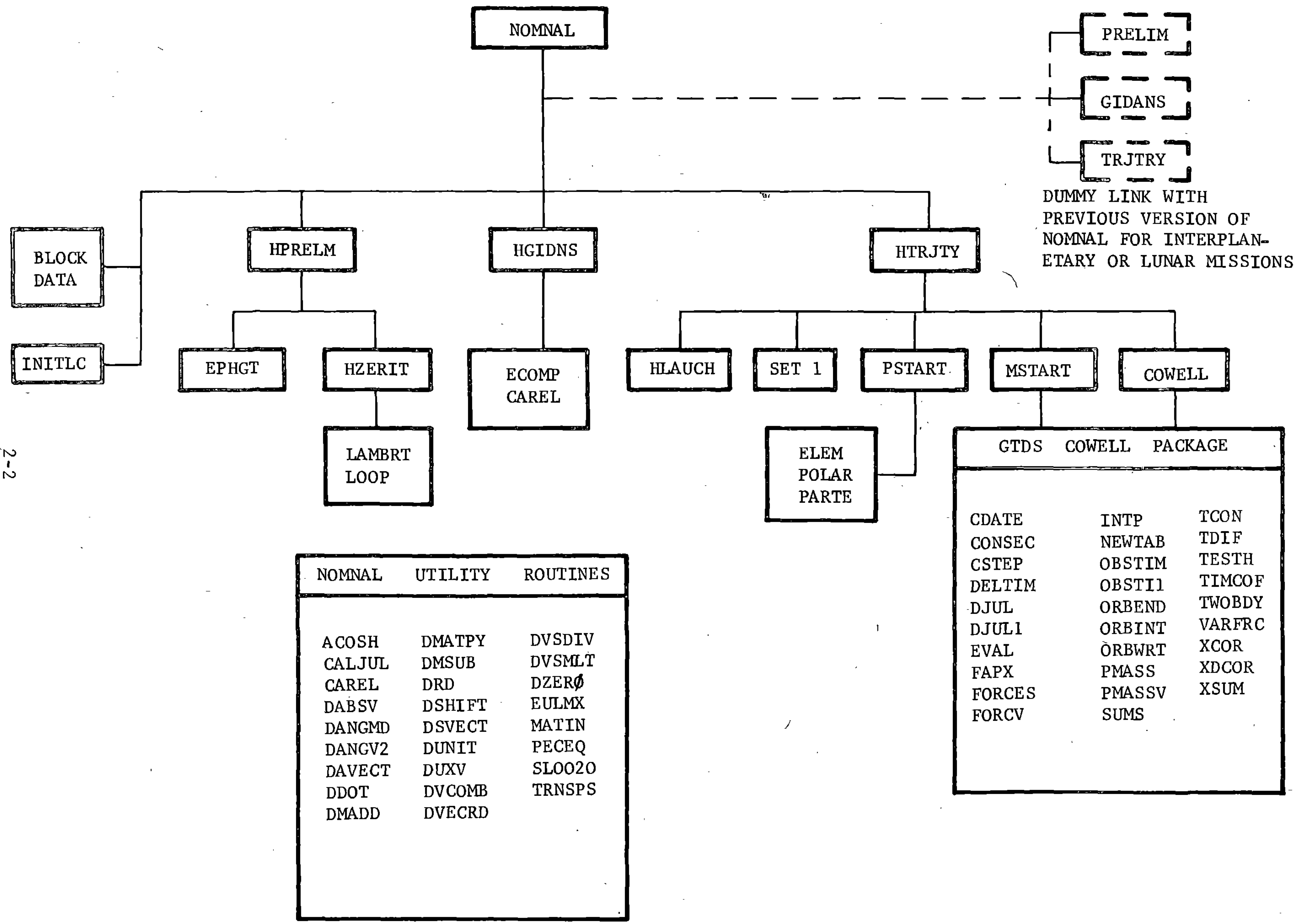


Figure 2.1 NOMNAL SUBROUTINE HIERARCHY

2.3.1 COMMON BLOCKS IN ALPHABETICAL ORDER

NAME(DIM)	DISP	DEFINITION
BDATA (SIZE 10)		
NBOD	0	NUMBER OF GRAVITATING BODIES
NB(3)	4	VECTOR OF BODIES (STEAP CONVENTION)
R		
FLAGS1 (SIZE 10)		
IBODY	0	CENTRAL BODY NUMBER (GTDS CONVENTION)
INDX4	4	NOT USED
INDY4	8	NOT USED
NEQ	C	NUMBER OF VARIATIONAL EQUATIONS
FLAGS2 (SIZE E8)		
ICENTB	0	CENTRAL BODY NUMBER (GTDS CONVENTION)
NTSFQS	4	TOTAL NUMBER OF SECTIONS
NSECTN	8	CURRENT SECTION NUMBER
INDSEC(10,3)	C	FLAGS FOR SECTION
IND(25)	84	FLAGS FOR CURRENT SECTION
FLAGS3 (SIZE 30)		
NCNM	0	NUMBER OF C(N,M) TO BE ESTIMATED
NSNM	4	NUMBER OF S(N,M) TO BE ESTIMATED
NG	8	TOTAL NUMBRE OF POTENTIAL COEFFICIENTS TO BE ESTIMATED
NSTATE	C	NUMBER OF STATE UNKNOWNS
KSTATE(6)	10	VECTOR OF LABEL NUMBERS FOR STATE UNKNOWNS
NCONDT	28	CONDITION NUMBER
MANDON	2C	MANEUVER OCCURENCE FLAG
FLAGS4 (SIZE 8)		
NCURS	0	GTDS PARAMETER
NENTRY	4	GTDS PARAMETER

NAME(DIM)	DISP	DEFINITION
FLAGS 5 (SIZE 4)		
NPS	0	GTDS PARAMETER
FLAGS6 (SIZE 8)		
ICENT	0	NUMBER OF CENTRAL BODY (GTDS CONVENTION)
ISUN	4	SUN-CENTRAL BODY FLAG
FLAGS7 (SIZE 10)		
NBOPT(3)	0	ARRAY OF GRAVITATING BODIES (GTDS CONVENTION)
NB	C	NUMBER OF BODIES
FLAGS9 (SIZE 4)		
NSTR	0	MAX NUMBER OF ACCELERATION POINTS
FSUNLT (SIZE 8)		
FSUNLT	0	OCCULTATION FLAG

NAME(DIM)	DISP	DEFINITION

GDATA1 (SIZE A8)

XL(6)	0	STATE VECTOR OF S/C AT LIBRATION POINT
DL	30	JULIAN DATE OF S/C AT LIBRATION POINT
XB(6)	38	STATE VECTOR OF S/C AT BURN INITIATION
TB	68	BURN TIME
XF(6)	70	STATE VECTOR OF S/C AT TCA
TDUR	A0	FLIGHT TIME

GDATA2 (SIZE 2D0)

BSTM(6,6)	0	STATE PARTIALS OVER BURN PHASE
ASTM(6,3)	120	BURN PARTIALS
STM(6,6)	180	STATE PARTIALS OVER COAST PHASE

GDATA3 (SIZE 18)

ACCTH(3)	0	BURN PARTIALS AT BURN EVENT INITIATION

GDATA4 (SIZE 18)

VOLP(3)	0	INERTIAL VELOCITY OF LIBRATION POINT

GFLAG (SIZE 18)

IBURN	0	FLAG TO INDICATE FINITE BURN INTEGRATION
INCFLG	4	NOT USED
ITOL	8	FLAG TO INDICATE TARGETING CONVERGENCE
ITMAX	C	MAXIMUM NUMBER OF TARGETING ITERATION
ITER	10	CURRENT TARGETING ITERATION
IBTYPE	14	FLAG TO INDICATE TYPE OF BURN STRATEGY

GTAR (SIZE 40)

DTAR(3)	0	TARGET VECTOR OF DESIRED VALUES
DTOL(3)	18	TARGET VALUE TOLERANCE VECTOR
PERT(2)	30	PERTURBATION VECTOR

NAME(DIM)	DISP	DEFINITION
HLDATA (SIZE 44)		
ZBIAS(6)	0	VECTOR ADDED TO LIBRATION STATE FOR TARGETING
RLIBR(2)	30	EARTH STATE SCALING CONSTANTS REQUIRED TO GENERATE STATE VECTOR OF LIBRATION POINT
LIBR	40	LIBRATION POINT OF INTEREST

NAME(DIM)	DISP	DEFINITION
IOFLAG (SIZE 338)		
IDON	0	FLAG TO INDICATE ORBIT FILE IS BEING WRITTEN ON THIS PASS
IDISK	4	FLAG TO INDICATE IF AN ORBIT FILE IS TO BE WRITTEN FOR THIS CASE
NCPR	8	NOT USED
NPOINT	C	NUMBER OF SPECIAL PRINT POINTS
TP	10	VECTOR OF SPECIAL PRINT POINTS
TMPR	330	NOMINAL PRINT INTERVAL

NAME(DIM)	DISP	DEFINITION
KNSTN2 (SIZE 28)		
PI	0	PI
PI2	8	TWO PI
RPD	10	RADIANS PER DEGREE
SPD	18	SECONDS PER DAY
XKMPAU	20	KILOMETERS PE ASTRONOMICAL UNIT

NAME(DIM)	DISP	DEFINITION
KNSTN3 (SIZE 108)		
SMU(11)	0	VECTOR OF GRAVITATIONAL CONSTANTS (STEAP CONVENTION)
RSOI(11)	58	RADIUS OF SPHERE OF INFLUENCE (STEAP CONVENTION)
RP(11)	80	RADIUS OF PLANETS (STEAP CONVENTION)

NAME(DIM)	DISP	DEFINITION
LPROF (SIZE 8)		
LAUNCH	0	FLAG TO INDICATE TARGET-LAUNCH COMPATIBILITY
LNON	4	FLAG TO INDICATE THAT LAUNCH PROFILE IS TO BE GENERATED ON THIS PASS

NAME(DIM)	DISP	DEFINITION
LAGS10 (SIZE 8)		
NOCOWL	0	NUMBER OF TIMES CSTEP WAS CALLED
ITERS	4	TOTAL NUMBER OF CORRECTOR ITERATIONS

NAME(DIM)	DISP	DEFINITION
LAGS11 (SIZE 1C)		
N1	0	NUMBER OF TERMS IN INTEGRATION FORMULAS FOR POSITION AND VELOCITY
N2	4	NUMBER OF TERMS IN INTEGRATION FORMULAS FOR PARTIAS
MAXIT(5)	8	MAXIMUM NUMBER OF ITERATIONS USED BY INTEGRATOR

NAME(DIM)	DISP	DEFINITION
LAGS12 (SIZE 4)		
K	0	CURRENT ACCELERATION POINT

NAME(DIM)	DISP	DEFINITION
LAGS13 (SIZE 8)		
NOFC(2)	0	NUMBER OF TIMES FROCES WERE REQUESTED

NAME(DIM)	DISP	DEFINITION
FLAGS14 (SIZE 8)		
NOSTEP(2)	0	GTDS COUNTER

NAME(DIM)	DISP	DEFINITION
LAGS15 (SIZE 4)		
IELEVN	0	ACCUMULATED ACCELERATION POINTS TO BE WRITTEN TO THE ORBIT FILE

NAME(DIM)	DISP	DEFINITION
LDATA (SIZE 50)		
FI	0	NOT USED
PSI1	8	FIRST INJECTION BURN ARC
PSI2	10	SECOND INJECTION BURN ARC
TIM1	18	DURATION OF 1ST INJECTION BURN
TIM2	20	DURATION OF 2ND INJECTION BURN
THELS	28	LONGITUDE OF LAUNCH SITE
PHILS	30	LATITUDE OF LAUNCH SITE
THEDOT	38	ROTATION RATE OF LAUNCH PLANET
RPRAT	40	INVERSE OF PARKING ORBIT RATE
SIGMAL	48	DESIRED LAUNCH AZIMUTH
RETRO	50	FLAG TO INDICATE DIRECTION OF MOTION
KOAST	58	NOT USED
LINK11 (SIZE 10)		
H	0	SIGNED STEP SIZE (SEC)
T	8	TIME UP TO WHICH INTEGRATION HAS PROGRESSED
LINK12 (SIZE 18)		
SECTIM	0	VECTOR OF TIME WITH SECTION
LINK14 (SIZE 8)		
RM	0	RADIUS MAGNITUDE
LINK19 (SIZE 50)		
TOL(10)	0	VECTOR OF TOLERANCES USED BY INTEGRATOR
LINK23 (SIZE 160)		
ALPHA(11)	0	PREDICTOR COEFFICIENT FOR POSITION TERMS
ALPHAS(11)	58	CORRECTOR COEFFICIENT FOR POSITION
ALPHAB(11)	80	PREDICTOR COEFFICIENT FOR POSITION PARTIAL TERMS
BETAB(11)	108	PREDICTOR COEFFICIENT FOR VELOCITY PARTIAL TERMS

NAME(DIM)	DISP	DEFINITION
LINK28		
CETOL	0	TRUNCATION ERROR TOLERANCE
LINK29		
XB(6,2)	0	ARRAY OF STATE VECTORS OF NON-CENTRAL BODIES WRT CENTRAL BODY
LINK33 (SIZE 6E0)		
BCS(11,10)	0	STARTER CORRECTOR COEFFICIENTS FOR POSITION
ACS(11,10)	370	STARTER CORRECTOR COEFFICIENTS FOR VELOCITY
LINK35 (SIZE B0)		
SACS(11)	0	STARTER COEFFICIENTS FOR FIRST SUM
SBCS(11)	58	STARTER COEFFICIENTS FOR SECOND SUM
LINK36 (SIZE 210)		
X1(11,3)	0	ARRAY OF STARTER POSITION VECTORS
X1D(11,3)	108	ARRAY OF STARTER VELOCITY VECTORS
LINK37 (SIZE 60)		
ACCB(3)	0	ACCELERATION DUE TO CENTRAL BODY
ACNB(3)	18	ACCELERATION DUE TO NON-CENTRAL BODIES
ACTH(3)	30	ACCELERATION DUE TO THRUST
ANCF(3)	48	ACCELERATION DUE TO THRUST AND NON-CENTRAL BODIES
LINK38 (SIZE 3F0)		
SX1(3)	0	1ST SUM FOR EQUATIONS OF MOTION
SX2(3)	18	2ND SUM FOR EQUATIONS OF MOTION
SV1(3,20)	30	1ST SUM FOR VARIATIONAL EQUATIONS
SV2(3,20)	210	2ND SUM FOR VARIATIONAL EQUATIONS

NAME(DIM)	DISP	DEFINITION
LINK39 (SIZE 10)		
YMDIC	0	YEAR, MONTH, DAY OF INITIAL CONDITIONS
HMSIC	8	HOUR, MINUTE, SECONDS OF INITIAL CONDITIONS
LINK40 (SIZE 1F8)		
XOLD(3)	0	PREVIOUS POSITION AND VELOCITY
YOLD(3,20)	18	PREVIOUS POSITION AND VELOCITY PARTIALS
LINK41 (SIZE 30)		
SPV(3)	0	INITIAL PREDICTED POSITION
SPC(3)	18	FINAL CORRECTED POSITION
LINK42 (SIZE 80)		
BETA(11)	0	PREDICTOR COEFFICIENT FOR VELOCITY TERMS
BETAS(11)	58	CORRECTOR COEFFICIENT FOR VELOCITY TERMS
LINK43 (SIZE 90)		
ACCBC(3,3)		CENTRAL BODY ACCELERATION FOR VARIATIONAL TERMS
ANCBV(3,3)		NON-CENTRAL BODY ACCELERATION FOR VARIATIONAL TERMS
LINK44 (SIZE 108)		
RM2	0	RADIUS MAGNITUDE SQUARED
RM3	8	RADIUS MAGNITUDE CUBED
GMBM3(3)	10	GRAVITATIONAL CONSTANT
BP(6,3)	28	ARRAY OF POSITIONS OF NON-CENTRAL BODIES WRT S/C
BPM2(3)	B8	SQUARE OF DISTANCE BETWEEN NON-CENTRAL BODIES AND S/C
BPM(3)	D0	DISTANCE BETWEEN NON-CENTRAL BODIES AND S/C
GMBPM3(3)	E8	GRAVITATIONAL CONSTANT
GMRM3	100	GRAVITATIONAL CONSTANT

NAME(DIM)	DISP	DEFINITION
MATRIX (SIZE 1C8)		
A(3,3)	0	MATRIX TO CONVERT SELENOCENTRIC TO SELENOGRAPHIC
ADOT(3,3)	48	DERIVATIVE OF MATRIX 'A'
B(3,3)	90	MATRIX TO CONVERT EARTH INERTIAL TO EARTH BODY FIXED
C(3,3)	D8	MATRIX TO CONVERT MEAN 1950.0 TO TRUE OF DATE COORDINATES
GHA	120	GREENWICH HOUR ANGLE
DXQ(18)	128	NOT USED
XP	1B8	X POLAR MOTION ANGLE
YP	1C0	Y POLAR MOTION ANGLE
MECEQ (SIZE 48)		
ECEQ(3,3)	0	ECLIPTIC TO EARTH EQUATORIAL ROTATION MATRIX
MEQEC (SIZE 48)		
EQEC(3,3)	0	EARTH EQUATORIAL TO ECLIPTIC ROTATION MATRIX
MLINK (SIZE 8)		
IPRE	0	FLAG TO INDICATE INITIAL RUN
KWIT	4	FLAG TO INDICATE TERMINATION OF CURRENT CASE
RLINK9 (SIZE 60)		
TWOPI	0	2.0 * PI
GM(11)	8	GRAVITATIONAL CONSTANTS OF CENTRAL BODIES (GTDS NUMBERING CONVENTION)
MSGS (SIZE 4)		
MSGLVL	0	FLAG TO INDICATE PRINTING OF DEBUG DATA
NEWLK1 (SIZE 28)		
BPP(3)	0	VECTOR FROM SUN TO SPACECRAFT
VBP	18	MAGNITUDE OF VECTOR BPP
BRAD	20	SOLAR RADIATION CONSTANT

NAME(DIM)	DISP	DEFINITION

NEWLK2 (SIZE 10)

SCAREA	0	AREA OF SPACECRAFT
CSUBR	8	SPACECRAFT REFLECTIVITY CONSTANT

PRT (SIZE 88)

MONTH(12)	0	HOLLERITH NAME OF MONTHS
PLANET(11)	30	HOLLERITH NAME OF PLANETS

RLINK4 (SIZE 130)

PVINT(6)	0	S/C INITIAL STATE VECTOR
AEINT(6)	30	KEPLERIAN ELEMENTS
SPINT(6)	60	SPHERICAL ELEMENTS
OBLINT(20)	90	AUXILLARY ELEMENTS

RLINK5 (SIZE 52B0)

X(3)	0	S/C POSITION VECTOR
XD(3)	18	S/C VELOCITY VECTOR
XDD(40,3)	30	ARRAY OF S/C ACCELERATION VECTORS
XV(3,20)	3F0	S/C POSITION PARTIALS
XVD(3,20)	5D0	S/C VELOCITY PARTIALS
XVDD(40,3,20)	7B0	ARRAY OF S/C ACCELERATION PARTIALS

RLINK6 (SIZE 270)

PX(3,3)	0	ACCELERATION PARTIALS WRT POSITION
PXD(3,3)	48	ACCELERATION PARTIALS WRT VELOCITY
ACCPAR(3,20)	90	ACCELERATION PARTIALS WRT PARAMETERS

SLPOPT (SIZE 38)

DJ	0	EVAL INTERNAL
IDAY1	8	DAY OF FIRST RECORD ON EPHEMERIS FILE
IYEAR	C	EVAL INTERNAL
ISPAN	10	EVAL INTERNAL
NBEPM(3)	14	BODIES FOR POLYNOMIAL COEFFICIENTS
NDEGRE(3)	20	DEGREE OF POLYNOMIALS
NCFDAY	2C	NUMBER OF DAYS PER CURVE FIT
ISLP50	30	EVAL INTERNAL
NBSLP	34	EVAL INTERNAL

NAME(DIM)	DISP	DEFINITION

SLPREC (SIZE D7C)

NAME(DIM)	DISP	DEFINITION
TSEC	0	TIME IN SECONDS FROM START OF THIS YEAR TO MIDPOINT OF THIS RECORD TIME INTERVAL
PPOLY(3,20,2)	8	POLYNOMIAL COEFFICIENTS FOR THE POSITION COORDINATES OF THE TWO BODIES
VPOLY(3,20,2)	3C8	POLYNOMIAL COEFFICIENTS FOR THE VELOCITY COORDINATES OF THE TWO BODIES
APOLY(3,3,10)	788	POLYNOMIAL COEFFICIENTS FOR THE 'A' MATRIX
CPOLY(3,3,10)	A58	POLYNOMIAL COEFFICIENTS FOR THE 'C' MATRIX
PDELH(10)	D28	POLYNOMIAL COEFFICIENTS FOR DELTA H
IDAY	D78	BEGINNING DAY OF THIS RECORD

THRUST (SIZE 60)

NAME(DIM)	DISP	DEFINITION
THRMAG	0	THRUST MAGNITUDE
XISP	8	SPECIFIC IMPULSE
SCMASS	10	S/C MASS
ALPHA	18	RIGHT ASCENSION OF BURN VECTOR
BETA	20	DECLINATION OF BURN VECTOR
TBURN	28	BURN DURATION
DMASS	30	MASS RATE
CURMAS	38	S/C MASS AT CURRENT INTEGRATION TIME
COSA	40	COSINE OF 'ALPHA'
COSB	48	COSINE OF 'BETA'
SINA	50	SINE OF 'ALPHA'
SINB	58	SINE OF 'BETA'

TIMCOF (SIZE E18)

NAME(DIM)	DISP	DEFINITION
COEF(72,6)	0	TIME DIFFERENCE COEFFICIENTS
POCOF(44,8)	6C0	POLAR MOTION COEFFICIENTS
JARG(72)	C40	JULIAN DATES FOR TIME DIFFERENCE COEFFICIENTS
IARG(44)	D60	JULIAN DATE INTERVALS FOR POLAR MOTION COEFFICIENTS
NDAYS	E10	SIZE OF VECTOR JARG
NNDAYS	E14	SIZE OF VECTOR IARG

USUN (SIZE 18)

NAME(DIM)	DISP	DEFINITION
USUN	0	UNIT VECTOR FROM CENTRAL BODY TO SUN

NAME(DIM)	DISP	DEFINITION
ZDATA (SIZE C)		
ATRY	0	INITIAL GUESS FOR SOLUTION TO LAMBERTS PROBLEM
NFR	8	NUMBER OF FULL REVOLUTION ABOUT THE EARTH PRIOR TO ENCOUNTER

2.3.2 COMMON VARIABLES IN ALPHABETICAL ORDER

VARIABLE(DIM)	BLOCK	DEFINITION
A(3,3)	MATRIX	MATRIX TO CONVERT SELENOCENTRIC TO SELENOGRAPHIC
ACCR(3)	LINK37	ACCELERATION DUE TO CENTRAL BODY
ACNB(3)	LINK37	ACCELERATION DUE TO NON-CENTRAL BODIES
ACCBC(3,3)	LINK43	CENTRAL BODY ACCELERATION FOR VARIATIONAL TERMS
ACCPAR(3,20)	RLINK6	ACCELERATION PARTIALS WRT PARAMETERS
ACCTH(3)	GDATA3	BURN PARTIALS AT BURN EVENT INITIATION
ACS(11,10)	LINK33	STARTER CORRECTOR COEFFICIENTS FOR VELOCITY
ACTH(3)	LINK37	ACCELERATION DUE TO THRUST
ADOT(3,3)	MATRIX	DERIVATIVE OF MATRIX 'A'
AEINT(6)	RLINK4	KEPLERIAN ELEMENTS
ALPHA	THRUST	RIGHT ASCENSION OF BURN VECTOR
ALPHA(11)	LINK23	PREDICTOR COEFFICIENT FOR POSITION TERMS
ALPHAS(11)	LINK23	CORRECTOR COEFFICIENT FOR POSITION
ALPHAB(11)	LINK23	PREDICTOR COEFFICIENT FOR POSITION PARTIAL TERMS
ANCBV(3,3)	LINK43	NON-CENTRAL BODY ACCELERATION FOR VARIATIONAL TERMS
ANCF(3)	LINK37	ACCELERATION DUE TO THRUST AND NON-CENTRAL BODIES
APOLY(3,3,10)	SLPREC	POLYNOMIAL COEFFICIENTS FOR THE 'A' MATRIX
ASTM(6,3)	GDATA2	BURN PARTIALS
ATRY	ZDATA	INITIAL GUESS FOR SOLUTION TO LAMBERTS PROBLEM
B(3,3)	MATRIX	MATRIX TO CONVERT EARTH INERTIAL TO EARTH BODY FIXED
BCS(11,10)	LINK33	STARTER CORRECTOR COEFFICIENTS FOR POSITION
BETA	THRUST	DECLINATION OF BURN VECTOR
BETA(11)	LINK42	PREDICTOR COEFFICIENT FOR VELOCITY TERMS
BETAB(11)	LINK23	PREDICTOR COEFFICIENT FOR VELOCITY PARTIAL TERMS
BETAS(11)	LINK42	CORRECTOR COEFFICIENT FOR VELOCITY TERMS
BP(6,3)	LINK44	ARRAY OF POSITIONS OF NON-CENTRAL BODIES WRT S/C
BPM(3)	LINK44	DISTANCE BETWEEN NON-CENTRAL BODIES AND S/C
BPM2(3)	LINK44	SQUARE OF DISTANCE BETWEEN NON-CENTRAL BODIES AND S/C
BPP(3)	NEWLK1	VECTOR FROM SUN TO SPACECRAFT
BRAD	NEWLK1	SOLAR RADIATION CONSTANT
BSTM(6,6)	GDATA2	STATE PARTIALS OVER BURN PHASE
C(3,3)	MATRIX	MATRIX TO CONVERT MEAN 1950.0 TO TRUE OF DATE COORDINATES
CETOL	LINK28	TRUNCATION ERROR TOLERANCE
COEF(72,6)	TIMCOF	TIME DIFFERENCE COEFFICIENTS
COSA	THRUST	COSINE OF 'ALPHA'
COSB	THRUST	COSINE OF 'BETA'
CPOLY(3,3,10)	SLPREC	POLYNOMIAL COEFFICIENTS FOR THE 'C' MATRIX
CSUBR	NEWLK2	SPACECRAFT REFLECTIVITY CONSTANT
CURMAS	THRUST	S/C MASS AT CURRENT INTEGRATION TIME
DJ	SLPOPT	EVAL INTERNAL
DL	GDATA1	JULIAN DATE OF S/C AT LIBRATION POINT
DMASS	THRUST	MASS RATE
DTAR(3)	GTAR	TARGET VECTOR OF DESIRED VALUES
DTOL(3)	GTAR	TARGET VALUE TOLERANCE VECTOR
DXQ(18)	MATRIX	NOT USED
ECEQ(3,3)	MECEQ	ECLIPTIC TO EARTH EQUATORIAL ROTATION MATRIX
EQEC(3,3)	MEQEC	EARTH EQUATORIAL TO ECLIPTIC ROTATION MATRIX

VARIABLE(DIM)	BLOCK	DEFINITION
FI	LDATA	NOT USED
FSUNLT	FSUNLT	OCCULTATION FLAG
GHA	MATRIX	GREENWICH HOUR ANGLE
GM(11)	RLINK9	GRAVITATIONAL CONSTANTS OF CENTRAL BODIES (GTDS NUMBERING CONVENTION)
GMBM3(3)	LINK44	GRAVITATIONAL CONSTANT
GMBPM3(3)	LINK44	GRAVITATIONAL CONSTANT
GMRM3	LINK44	GRAVITATIONAL CONSTANT
H	LINK11	SIGNED STEP SIZE (SEC)
HMSIC	LINK39	HOUR, MINUTE, SECONDS OF INITIAL CONDITIONS
IARG(44)	TIMCOF	JULIAN DATE INTERVALS FOR POLAR MOTION COEFFICIENTS
IBODY	FLAGS1	CENTRAL BODY NUMBER (GTDS CONVENTION)
IBTYPE	GFLAG	FLAG TO INDICATE TYPE OF BURN STRATEGY
IBURN	GFLAG	FLAG TO INDICATE FINITE BURN INTEGRATION
ICENT	FLAGS6	NUMBER OF CENTRAL BODY (GTDS CONVENTION)
ICENTB	FLAGS2	CENTRAL BODY NUMBER (GTDS CONVENTION)
IDAY	SLPREC	BEGINNING DAY OF THIS RECORD
IDAY1	SLPOPT	DAY OF FIRST RECORD ON EPHEMERIS FILE
IDISK	IOFLAG	FLAG TO INDICATE IF AN ORBIT FILE IS TO BE WRITTEN FOR THIS CASE
IDON	IOFLAG	FLAG TO INDICATE ORBIT FILE IS BEING WRITTEN ON THIS PASS
IELEVN	LAGS15	ACCUMULATED ACCELERATION POINTS TO BE WRITTEN TO THE ORBIT FILE
INCFLG	GFLAG	NOT USED
IND(25)	FLAGS2	FLAGS FOR CURRENT SECTION
INDSEC(10,3)	FLAGS2	FLAGS FOR SECTION
INDX4	FLAGS1	NOT USED
INDY4	FLAGS1	NOT USED
ISLP50	SLPOPT	EVAL INTERNAL
ISPAN	SLPOPT	EVAL INTERNAL
ISUN	FLAGS6	SUN-CENTRAL BODY FLAG
ITER	GFLAG	CURRENT TARGETING ITERATION
ITERS	LAGS10	TOTAL NUMBER OF CORRECTOR ITERATIONS
ITMAX	GFLAG	MAXIMUM NUMBER OF TARGETING ITERATION
ITOL	GFLAG	FLAG TO INDICATE TARGETING CONVERGENCE
IYEAR	SLPOPT	EVAL INTERNAL
JARG(72)	TIMCOF	JULIAN DATES FOR TIME DIFFERENCE COEFFICIENTS
K	LAGS12	CURRENT ACCELERATION POINT
KOAST	LDATA	NOT USED
KSTATE(6)	FLAGS3	VECTOR OF LABEL NUMBERS FOR STATE UNKNOWNS
KWIT	MLINK	FLAG TO INDICATE TERMINATION OF CURRENT CASE
LAUNCH	LPROF	FLAG TO INDICATE TARGET-LAUNCH COMPATIBILITY
LIBR	HLDATA	LIBRATION POINT OF INTEREST
LNON	LPROF	FLAG TO INDICATE THAT LAUNCH PROFILE IS TO BE GENERATED ON THIS PASS
MANDON	FLAGS3	MANEUVER OCCURENCE FLAG
MAXIT(5)	LAGS11	MAXIMUM NUMBER OF ITERATIONS USED BY INTEGRATOR
MONTH(12)	PRY	HOLLERITH NAME OF MONTHS
MSGLVL	MSGS	FLAG TO INDICATE PRINTING OF DEBUG DATA

VARIABLE(DIM)	BLOCK	DEFINITION
NB	FLAGS7	NUMBER OF BODIES
NB(3)	BDATA	VECTOR OF BODIES (STEAP CONVENTION)
NBEPM(3)	SLPOPT	BODIES FOR POLYNOMIAL COEFFICIENTS
NBSLP	SLPOPT	EVAL INTERNAL
NBOD	BDATA	NUMBER OF GRAVITATING BODIES
NBOPT(3)	FLAGS7	ARRAY OF GRAVITATING BODIES (GTDS CONVENTION)
NCFDAY	SLPOPT	NUMBER OF DAYS PER CURVE FIT
NCNM	FLAGS3	NUMBER OF C(N,M) TO BE ESTIMATED
NCONDT	FLAGS3	CONDITION NUMBER
NCPR	IOFLAG	NOT USED
NCURS	FLAGS4	GTDS PARAMETER
NDAYS	TIMCOF	SIZE OF VECTOR JARG
NDEGRE(3)	SLPOPT	DEGREE OF POLYNOMIALS
NENTRY	FLAGS4	GTDS PARAMETER
NEQ	FLAGS1	NUMBER OF VARIATIONAL EQUATIONS
NFR	ZDATA	NUMBER OF FULL REVOLUTION ABOUT THE EARTH PRIOR TO ENCOUNTER
NG	FLAGS3	TOTAL NUMBRE OF POTENTIAL COEFFICIENTS TO BE ESTIMATED
NNDAYS	TIMCOF	SIZE OF VECTOR IARG
NOCOWL	LAGS10	NUMBER OF TIMES CSTEP WAS CALLED
NOFC(2)	LAGS13	NUMBER OF TIMES FROCES WERE REQUESTED
NOSTEP(2)	LAGS14	GTDS COUNTER
NPOINT	IOFLAG	NUMBER OF SPECIAL PRINT POINTS
NPS	FLAGS5	GTDS PARAMETER
NSECTN	FLAGS2	CURRENT SECTION NUMBER
NSNM	FLAGS3	NUMBER OF S(N,M) TO BE ESTIMATED
NSTATE	FLAGS3	NUMBER OF STATE UNKNOWNS
NSTR	FLAGS9	MAX NUMBER OF ACCELERATION POINTS
NTSFQS	FLAGS2	TOTAL NUMBER OF SECTIONS
N1	LAGS11	NUMBER OF TERMS IN INTEGRATION FORMULAS FOR POSITION AND VELOCITY
N2	LAGS11	NUMBER OF TERMS IN INTEGRATION FORMULAS FOR PARTIAS
OBLINT(20)	RLINK4	AUXILLARY ELEMENTS
PDELH(10)	SLPREC	POLYNOMIAL COEFFICIENTS FOR DELTA H
PERT(2)	GTAR	PERTURBATION VECTOR
PHILS	LDATA	LATITUDE OF LAUNCH SITE
PI	KNSTN2	PI
PI2	KNSTN2	TWO PI
PLANET(11)	PRT	HOLLERITH NAME OF PLANETS
POCOF(44,8)	TIMCOF	POLAR MOTION COEFFICIENTS
PPOLY(3,20,2)	SLPREC	POLYNOMIAL COEFFICIENTS FOR THE POSITION COORDINATES OF THE TWO BODIES
PSI1	LDATA	FIRST INJECTION BURN ARC
PSI2	LDATA	SECOND INJECTION BURN ARC
PVINT(6)	RLINK4	S/C INITIAL STATE VECTOR
PX(3,3)	RLINK6	ACCELERATION PARTIALS WRT POSITION
PXD(3,3)	RLINK6	ACCELERATION PARTIALS WRT VELOCITY

VARIABLE(DIM)	BLOCK	DEFINITION
RETRO	LDATA	FLAG TO INDICATE DIRECTION OF MOTION
RLIBR(2)	HLDATA	EARTH STATE SCALING CONSTANTS REQUIRED TO GENERATE STATE VECTOR OF LIBRATION POINT
RM	LINK14	RADIUS MAGNITUDE
RM2	LINK44	RADIUS MAGNITUDE SQUARED
RM3	LINK44	RADIUS MAGNITUDE CUBED
RP(11)	KNSTN2	RADIUS OF PLANETS (STEAP CONVENTION)
RPD	KNSTN2	RADIANS PER DEGREE
RPRAT	LDATA	INVERSE OF PARKING ORBIT RATE
RSOI(11)	KNSTN2	RADIUS OF SPHERE OF INFLUENCE (STEAP CONVENTION)
SACS(11)	LINK35	STARTER COEFFICIENTS FOR FIRST SUM
SBCS(11)	LINK35	STARTER COEFFICIENTS FOR SECOND SUM
SCAREA	NEWLK2	AREA OF SPACECRAFT
SCMASS	THRUST	S/C MASS
SECTIM	LINK12	VECTOR OF TIME WITH SECTION
SIGMAL	LDATA	DESIRED LAUNCH AZIMUTH
SINA	THRUST	SINE OF 'ALPHA'
SINB	THRUST	SINE OF 'BETA'
SMU(11)	KNSTN3	VECTOR OF GRAVITATIONAL CONSTANTS (STEAP CONVENTION)
SPC(3)	LINK41	FINAL CORRECTED POSITION
SPD	KNSTN2	SECONDS PER DAY
SPINT(6)	RLINK4	SPHERICAL ELEMENTS
SPV(3)	LINK41	INITIAL PREDICTED POSITION
STM(6,6)	GDATA2	STATE PARTIALS OVER COAST PHASE
SV1(3,20)	LINK38	1ST SUM FOR VARIATIONAL EQUATIONS
SV2(3,20)	LINK38	2ND SUM FOR VARIATIONAL EQUATIONS
SX1(3)	LINK38	1ST SUM FOR EQUATIONS OF MOTION
SX2(3)	LINK38	2ND SUM FOR EQUATIONS OF MOTION
T	LINK11	TIME UP TO WHICH INTEGRATION HAS PROGRESSED
TB	GDATA1	BURN TIME
TBURN	THRUST	BURN DURATION
TDUR	GDATA1	FLIGHT TIME
THEDOT	LDATA	ROTATION RATE OF LAUNCH PLANET
THELS	LDATA	LONGITUDE OF LAUNCH SITE
THRMAG	THRUST	THRUST MAGNITUDE
TIM1	LDATA	DURATION OF 1ST INJECTION BURN
TIM2	LDATA	DURATION OF 2ND INJECTION BURN
TMPR	IOFLAG	NOMINAL PRINT INTERVAL
TSEC	SLPREC	TIME IN SECONDS FROM START OF THIS YEAR TO MIDPOINT OF THIS RECORD TIME INTERVAL
TOL(10)	LINK19	VECTOR OF TOLERANCES USED BY INTEGRATOR
TP	IOFLAG	VECTOR OF SPECIAL PRINT POINTS
TWOPI	RLINK9	2.0 * PI

VARIABLE(DIM)	BLOCK	DEFINITION
USUN	USUN	UNIT VECTOR FROM CENTRAL BODY TO SUN
VBP	NEWLK1	MAGNITUDE OF VECTOR BPP
VOLP(3)	GDATA4	INERTIAL VELOCITY OF LIBRATION POINT
VPOLY(3,20,2)	SLPREC	POLYNOMIAL COEFFICIENTS FOR THE VELOCITY COORDINATES OF THE TWO BODIES
X(3)	RLINK5	S/C POSITION VECTOR
XB(6)	GDATA1	STATE VECTOR OF S/C AT BURN INITIATION
XB(6,2)	LINK29	ARRAY OF STATE VECTORS OF NON-CENTRAL BODIES WRT CENTRAL BODY
XD(3)	RLINK5	S/C VELOCITY VECTOR
XDD(40,3)	RLINK5	ARRAY OF S/C ACCELERATION VECTORS
XF(6)	GDATA1	STATE VECTOR OF S/C AT TCA
XISP	THRUST	SPECIFIC IMPULSE
XKMPAU	KNSTN2	KILOMETERS PE ASTRONOMICAL UNIT
XL(6)	GDATA1	STATE VECTOR OF S/C AT LIBRATION POINT
XP	MATRIX	X POLAR MOTION ANGLE
XOLD(3)	LINK40	PREVIOUS POSITION AND VELOCITY
XV(3,20)	RLINK5	S/C POSITION PARTIALS
XVD(3,20)	RLINK5	S/C VELOCITY PARTIALS
XVDD(40,3,20)	RLINK5	ARRAY OF S/C ACCELERATION PARTIALS
XI(11,3)	LINK36	ARRAY OF STARTER POSITION VECTORS
XID(11,3)	LINK36	ARRAY OF STARTER VELOCITY VECTORS
YMDIC	LINK39	YEAR, MONTH, DAY OF INITIAL CONDITIONS
YOLD(3,20)	LINK40	PREVIOUS POSITION AND VELOCITY PARTIALS
YP	MATRIX	Y POLAR MOTION ANGLE
ZBIAS(6)	HLDATA	VECTOR ADDED TO LIBRATION STATE FOR TARGETING

3. ERRAN PROGRAM STRUCTURE

3.1 ERRAN PROGRAM DESCRIPTION

THE STEAP-L/ERRAN PROGRAM IS A DIGITAL COMPUTER PROGRAM WRITTEN IN FORTRAN IV COMPATIBLE WITH THE IBM 360/370 SYSTEM. THE PROGRAM CONTAINS ABOUT SIXTY SUBROUTINES OF WHICH ABOUT A TENTH ARE ASSOCIATED WITH THE COWELL FILE READER, ABOUT A SIXTH ARE GENERAL UTILITY ROUTINES, ABOUT A THIRD ARE COMMON TO MEASUREMENT AND EVENT PROCESSING, AND THE REST ARE SPECIFIC TO MEASUREMENT PROCESSING OR TO PARTICULAR EVENT PROCESSING. THE PROGRAM ACCESSES FIVE DATA SETS DURING OPERATION: THE STANDARD FORTRAN INPUT (SYSIN), THE PRESENTATION OUTPUT (SYSOUT=A) AND PUNCH OUTPUT (SYSOUT=B) DATA SETS, THE DIRECT ACCESS SOLAR/LUNAR/PLANETARY EPHEMERIS FILE USED BY THE GTDS (DSN=GTDS.SLP1950.JAN71), AND THE SEQUENTIAL ORBIT FILE ON WHICH THE TRAJECTORY AND STATE TRANSITION MATRIX DATA ARE STORED. THE (OVERLAID) PROGRAM REQUIRES ABOUT 310K BYTES STORAGE.

3.2 ERRAN SUBROUTINE HIERARCHY

FIGURE 3.1 ILLUSTRATES THE GENERAL PROGRAM STRUCTURE. THE MAIN PROGRAM CALLS TWO EXECUTIVE ROUTINES: DATA, FOR DATA INPUT AND INITIALIZATION, AND ERRAN, FOR MEASUREMENT AND EVENT PROCESSING. ERRAN CONTROLS MEASUREMENT PROCESSING DIRECTLY, BUT CALLS SPECIFIC (OVERLAID) SUBROUTINES TO PROCESS PARTICULAR EVENTS. ERRAN CALLS SETEVN FOR THE EIGENVECTOR EVENT, WHOSE PROCESSING IS COMMON TO ALL EVENTS. ERRAN THEN WILL CALL PRED IF PROCESSING A PREDICTION EVENT, OR WILL CALL GUIDM IF PROCESSING A GUIDANCE EVENT OR A FINAL INSERTION EVENT. ERRAN THEN MAY CALL GENGID IF PROCESSING A GUIDANCE EVENT WITH GENERALIZED COVARIANCES. CHAPTER 4 PROVIDES AN INDEX OF ALL SUBROUTINES USED IN ERRAN AND GIVES INDIVIDUAL ROUTINE DOCUMENTATION INCLUDING DESCRIPTIONS, ANALYSES, AND FLOWCHARTS.

3.3 ERRAN COMMON VARIABLES

THE FOLLOWING TWO SUBSECTIONS PROVIDE DEFINITIONS OF THE VARIABLES APPEARING IN COMMON BLOCKS IN THE ERRAN PROGRAMS. SUBSECTION 3.3.1 LISTS THE COMMON BLOCKS IN ALPHABETICAL ORDER, GIVES THE SIZE OF THE BLOCKS, AND DEFINES THE VARIABLES APPEARING IN EACH BLOCK. SUBSECTION 3.3.2 LISTS ALL THE COMMON VARIABLES IN ALPHABETICAL ORDER, GIVES THE BLOCK TO WHICH EACH BELONGS, AND DEFINES THE VARIABLES.

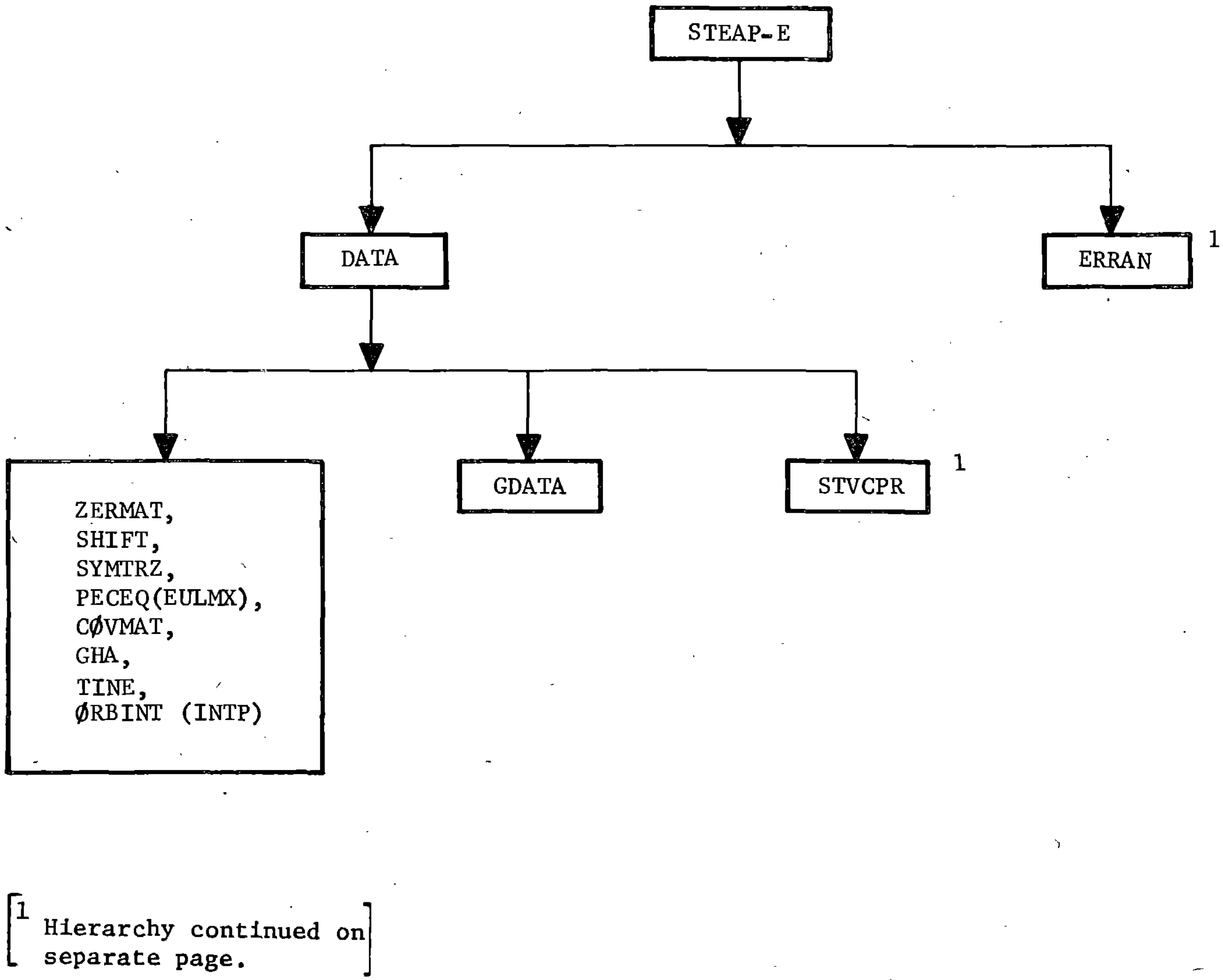
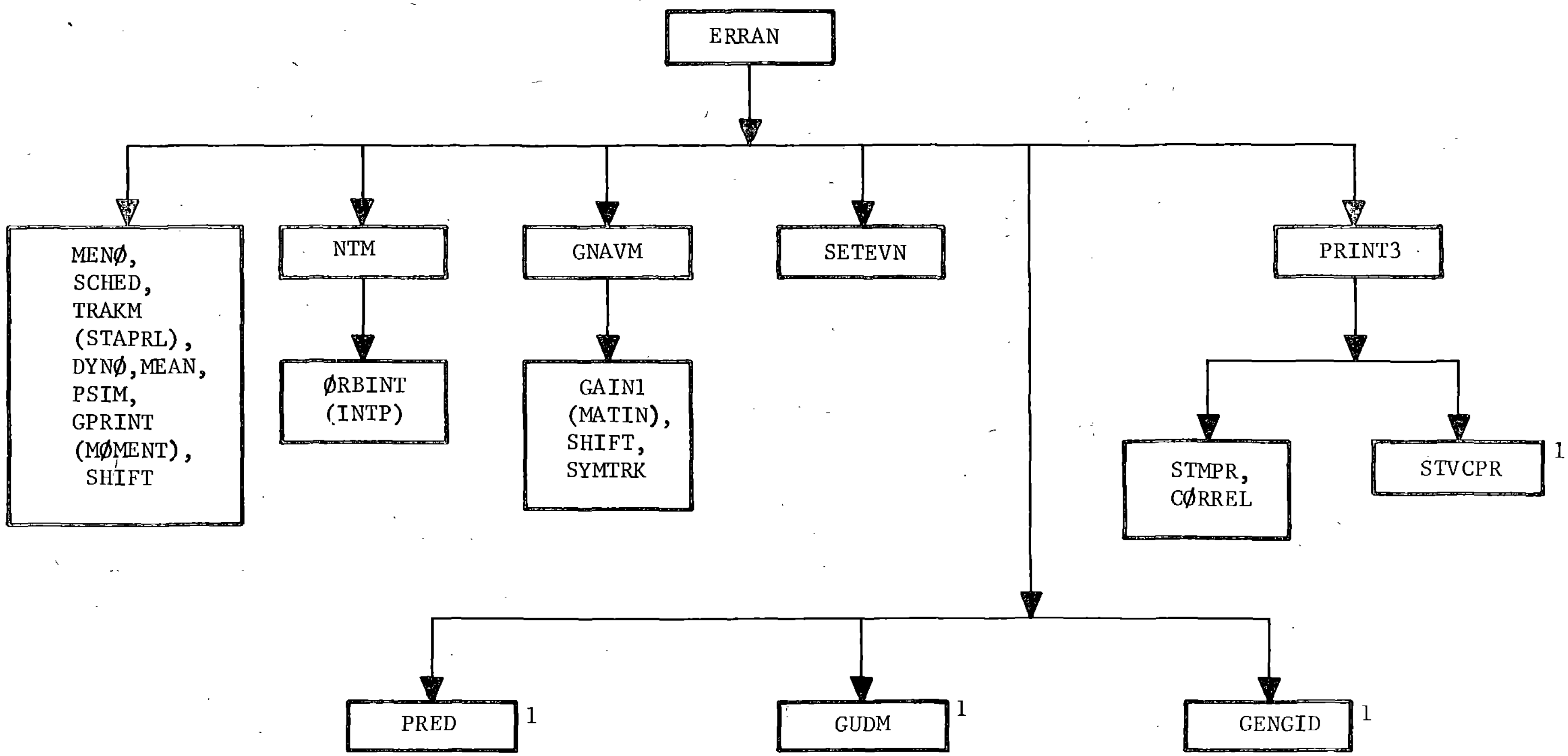
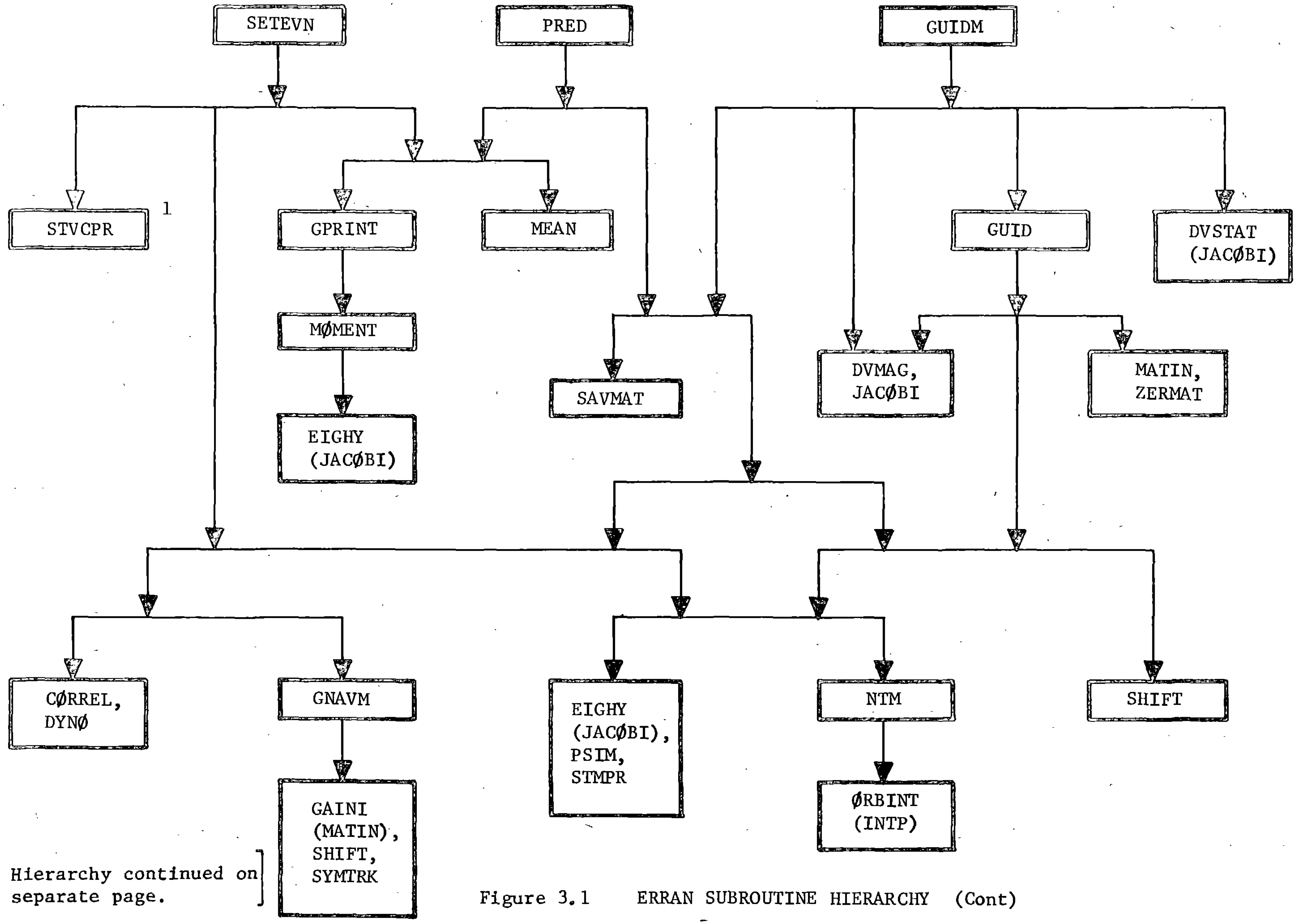


Figure 3.1 ERRAN SUBROUTINE HIERARCHY



1 Hierarchy continued on separate page.

Figure 3.1 ERRAN SUBROUTINE HIERARCHY (*Cont*)



[1 Hierarchy continued on separate page.]

Figure 3.1 ERRAN SUBROUTINE HIERARCHY (Cont)

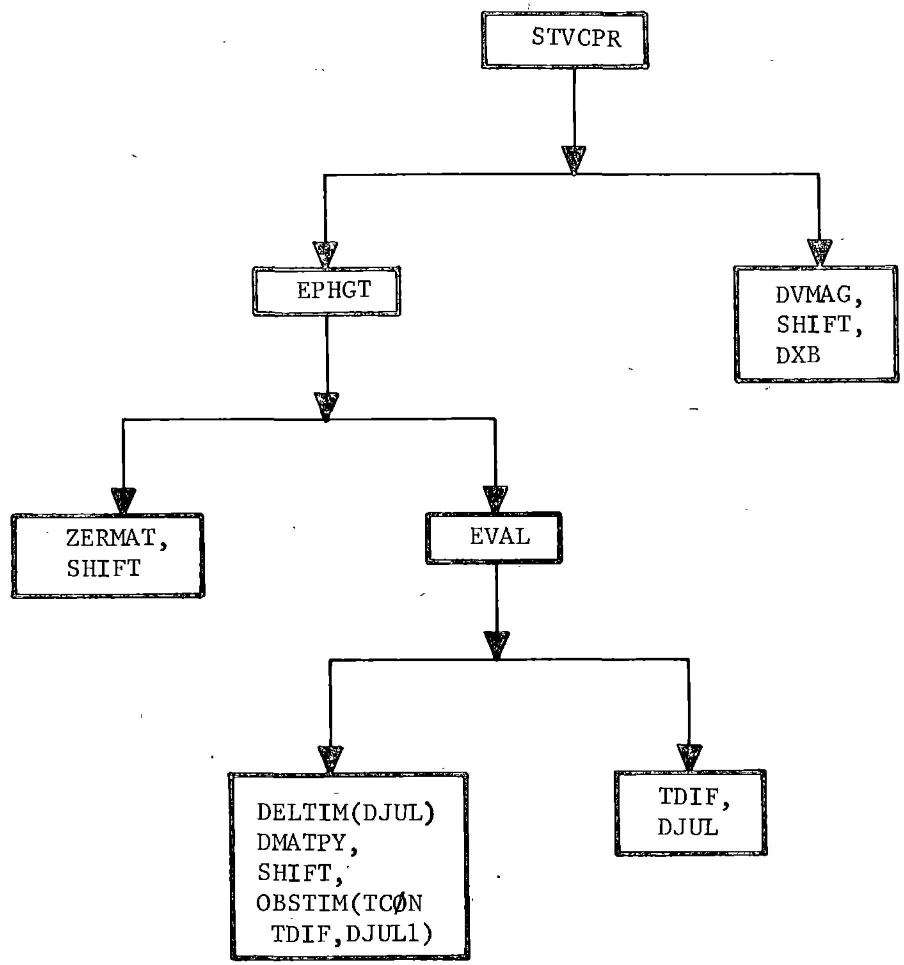


Figure 3.1 ERRAN SUBROUTINE HIERARCHY (Cont)

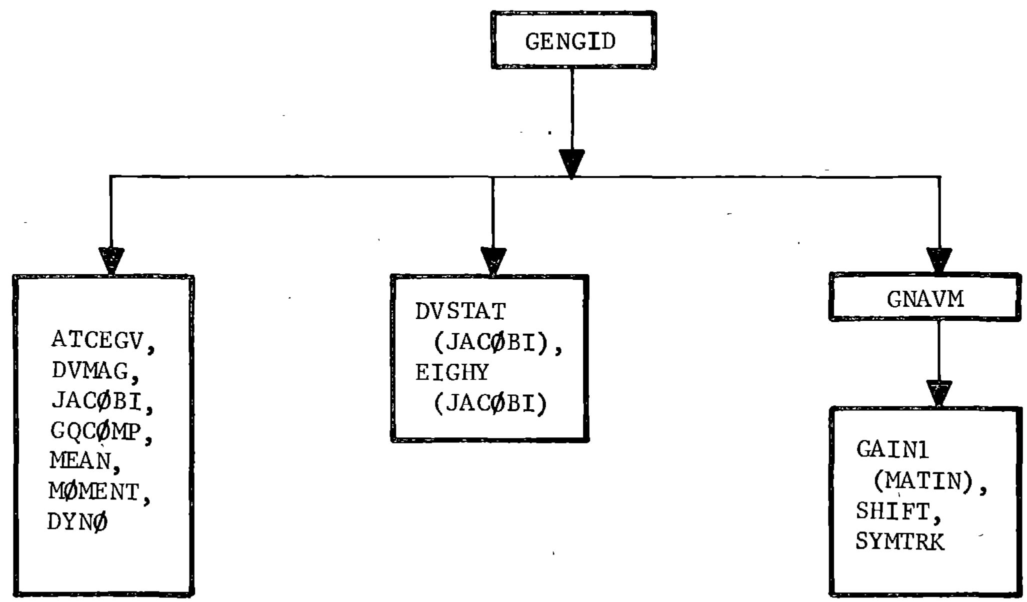


Figure 3.1 ERRAN SUBROUTINE HIERARCHY (Cont)

3.3.1 COMMON BLOCKS IN ALPHABETICAL ORDER

PREVIOUS VERSIONS OF STEAP REQUIRED THE FOLLOWING COMMON BLOCKS WHICH ARE NO LONGER NECESSARY: CONST3(SIZE 40), ENCKE(SIZE 4), PRELCM(SIZE 3C4), STMG(SIZE 4), TRAJCD(SIZE 30), TRJ(SIZE 1B0), AND UPDATE(SIZE 54).

NAME(DIM)	DISP	DEFINITION
BLK (SIZE 954)		
T	0	TRAJECTORY TIME IN DAYS
PMASS(11)	8	GRAVITATIONAL CONSTANTS (A.U.**3/DAY**2)
COMEG(4,9)	60	* NO LONGER USED
CINC(4,9)	180	* NO LONGER USED
COMEGT(4,9)	2A0	*** NO LONGER USED
SMJR(2,9)	3C0	* NO LONGER USED
CECC(4,9)	450	* NO LONGER USED
CMEAN(4,9)	570	* NO LONGER USED
MUPLAN(11)	690	GRAVITATIONAL CONSTANTS (KM((3/SEC**2)
CSAX(2,9)	6E8	NO LONGER USED
EMN(15)	778	NO LONGER USED
RADIUS(11)	7F0	RADII OF THE PLANETS (A.U.)
RMASS(11)	848	GRAVITATIONAL CONSTANTS (RELATIVE TO SUN)
SPHERE(11)	8A0	SPHERES OF INFLUENCE (A.U.)
XP(6)	8F8	STATE VECTOR OF PLANET
NO(11)	928	NO LONGER USED

NAME(DIM)	DISP	DEFINITION
COM (SIZE 40)		
PI	0	MATHEMATICAL CONSTANT PI
RAD	8	NUMBER OF DEGREES PER RADIAN
ITRAT	10	* NO LONGER USED
KOUNT	14	* NO LONGER USED
INCMNT	18	* NO LONGER USED
INCPR	1C	* NO LONGER USED
INC	20	*** NO LONGER USED
IPR	24	* NO LONGER USED
NBODYI	28	* NO LONGER USED
NBODY	2C	* NO LONGER USED
IPRT(4)	30	* NO LONGER USED

NAME(DIM)	DISP	DEFINITION

CONST (SIZE D4)

OMEGA	0	ROTATION RATE OF EARTH
EPS	8	EARTH OBLIQUITY
SAL(3)	10	STATION ALTITUDES (ABOVE RADIUS OF EARTH)
SLAT(3)	28	STATION LATITUDES
SLON(3)	40	STATION LONGITUDES
DNCN(3)	58	DYNAMIC NOISE CONSTANTS
MNCN(12)	70	MEASUREMENT NOISE CONSTANTS
NST	D0	NUMBER OF STATIONS TO BE USED (MAXIMUM 3)

CONST2 (SIZE 58)

UST(3)	0	X-DIRECTION COSINE FOR STARS
VST(3)	18	Y-DIRECTION COSINE FOR STARS
WST(3)	30	Z-DIRECTION COSINE FOR STARS
FOP	48	OFF-DIAGONAL ANNIHILATION VALUE (POSITION)
FOV	50	OFF-DIAGONAL ANNIHILATION VALUE (VELOCITY)

DPNUM (SIZE 88)

ZERO	0	DOUBLE-PRECISION VALUE OF ZERO (0.0)
ONE	8	DOUBLE-PRECISION VALUE OF ONE (1.0)
TWO	10	DOUBLE-PRECISION VALUE OF TWO (2.0)
HALF	18	DOUBLE-PRECISION VALUE OF HALF (0.5)
THREE	20	DOUBLE-PRECISION VALUE OF THREE (3.0)
EM1	28	DOUBLE-PRECISION VALUE OF 1.E-1
EM2	30	DOUBLE-PRECISION VALUE OF 1.E-2
EM3	38	DOUBLE-PRECISION VALUE OF 1.E-3
EM4	40	DOUBLE-PRECISION VALUE OF 1.E-4
EM5	48	DOUBLE-PRECISION VALUE OF 1.E-5
EM6	50	DOUBLE-PRECISION VALUE OF 1.E-6
EM7	58	DOUBLE-PRECISION VALUE OF 1.E-7
EM8	60	DOUBLE-PRECISION VALUE OF 1.E-8
EM9	68	DOUBLE-PRECISION VALUE OF 1.E-9
EP50	70	DOUBLE-PRECISION VALUE OF 1.E+50
TWOPI	78	DOUBLE-PRECISION VALUE OF 2.*PI
EM13	80	DOUBLE-PRECISION VALUE OF 1.E-13

NAME(DIM)	DISP	DEFINITION

EVENT (SIZE 4B4)

NAME(DIM)	DISP	DEFINITION
TEV(50)	0	SCHEDULED TIMES OF EVENTS
TPT2(20)	190	TIMES PREDICTED TO IN PREDICTION EVENTS
SIGRES	230	VARIANCE OF RESOLUTION ERROR
SIGPRO	238	VARIANCE OF PROPORTIONALITY ERROR
SIGALP	240	VARIANCE OF ERROR IN POINTING ANGLE 1
SIGBET	248	VARIANCE OF ERROR IN POINTING ANGLE 2
HP7	250	* NO LONGER USED
P7	258	* NO LONGER USED
TAU7	260	* NO LONGER USED
AINC7	268	*** NO LONGER USED
ANODE7	270	* NO LONGER USED
PERP7	278	* NO LONGER USED
ECC7	280	* NO LONGER USED
DV8(3)	288	* NO LONGER USED
BRNTIM	2A0	DURATION (DAYS) OF FINAL INSERTION BURN
NEV	2A8	NUMBER OF EVENTS SCHEDULED
IEVNT(50)	2AC	CODED EVENT TYPES CORRESPONDING TO REV TIMES
IHYP1	374	NO LONGER USED
IEIG	378	NO LONGER USED
ICDT3(20)	37C	CODES FOR GUIDANCE POLICIES
NPE	3CC	COUNT OF PREDICTION EVENTS
NGE	3D0	COUNT OF GUIDANCE EVENTS
IPOL	3D4	NO LONGER USED
IIPOL	3D8	NO LONGER USED
ICDQ3(20)	3DC	NO LONGER USED
NEV1	42C	NUMBER OF SCHEDULED EIGENVECTOR EVENTS
NEV2	430	NUMBER OF SCHEDULED PREDICTION EVENTS
NEV3	434	NUMBER OF SCHEDULED GUIDANCE EVENTS
NEV4	438	NUMBER OF SCHEDULED INSERTION EVENTS
NQE	436	NO LONGER USED
NEV5	440	NO LONGER USED
NEV6	444	NO LONGER USED
NAE	448	NO LONGER USED
NAF6(20)	44C	NO LONGER USED
NEV7	49C	NO LONGER USED
IOP7	4A0	NO LONGER USED
NEV8	4A4	NO LONGER USED
NEV9	4A8	NO LONGER USED
NEV10	4AC	NO LONGER USED
NEV11	4B0	NO LONGER USED

FLAGS2 (SIZE E8)

NAME(DIM)	DISP	DEFINITION
ICENTB	0	CENTRAL BODY NUMBER
NTSEQS	4	TOTAL NUMBER OF SECTIONS
NSECTN	8	CURRENT SECTION NUMBER
INDSEC(10,3)	C	SECTION FLAGS
IND(25)	84	CURRENT SECTION FLAGS

NAME(DIM)	DISP	DEFINITION

GAINC (SIZE 1628)

PMIN(6,6)	0	POS/VEL COVARIANCE BEFORE MEASUREMENT (WLS)
PSMIN(15,15)	120	SOLVE-FOR COVARIANCE BEFORE MEASUREMENT (WLS)
CMIN(6,15)	828	CORRELATION MATRIX, POS/VEL AND SOLVE-FORS
PPLU(6,6)	AF8	POS/VEL COVARIANCE AFTER MEASUREMENT (WLS)
PSPLU(15,15)	C18	SOLVE-FOR COVARIANCE AFTER MEASUREMENT (WLS)
CPLU(6,15)	1320	CORRELATION MATRIX, POS/VEL AND SOLVE-FORS
RSAVE(6)	15F0	STATE VECTOR AT TLAST
TLAST	1620	TIME WHEN MEASUREMENT LAST PROCESSED

GCA (SIZE E0)

XIG(15)	0	IGNORE PARAMETER LABLES
IAUGW(24)	78	IGNORE PARAMETER AUGMENTATION VECTOR
NDIM4	D8	DIMENSION OF IGNORE PARAMETER STATE
IGEN	DC	=0, PERFORM NO GENERALIZED COVARIANCE ANALYSIS =1, PERFORM GENERALIZED COVARIANCE ANALYSIS

GENGD (SIZE 40)

EE(4)	0	ACTUAL MEANS OF EXECUTION ERROR PARAMETERS
EEE(4)	20	VARIANCES OF EXECUTION ERROR PARAMETERS

GENGDI (SIZE 23E8)

		ACTUAL CONTROL SECOND MOMENT MATRICES
GPG(6,6)	0	STATE
GCXXSG(6,15)	120	STATE/SOLVE-FOR VECTOR
GCXUG(6,8)	3F0	NO LONGER USED
GCXVG(6,15)	570	STATE/MEASUREMENT CONSIDERS
GCXWG(6,15)	840	STATE/IGNORE PARAMETERS
GPSG(15,15)	B10	SOLVE-FOR VECTOR
GCXSUG(15,8)	1218	NO LONGER USED
GCXSVG(15,15)	15D8	SOLVE-FOR/MEASUREMENT CONSIDERS
GCXSWG(15,15)	1CE0	SOLVE-FOR/IGNORE PARAMETERS

NAME(DIM)	DISP	DEFINITION.
GENRL (SIZE 5708)		
GP(6,6)	0	ACTUAL STATE 2ND MOMENT MATRIX
GCXXS(6,15)	120	ACTUAL 2ND MOMENT MATRIX, STATE/SOLVE-FORS
GCXU(6,8)	3F0	NO LONGER USED
GCXV(6,15)	570	ACTUAL 2ND MOMENT MATRIX, STATE/MEAS. CONS.
GPS(15,15)	840	ACTUAL 2ND MOMENT MATRIX, SOLVE-FOR VECTOR
GCXSU(15,8)	F48	NO LONGER USED
GCXSV(15,15)	1308	ACTUAL 2ND MOMENT MATRIX, SOLVE-FOR/MS. CN.
GCXSW(15,15)	1A10	ACTUAL 2ND MOMENT MATRIX, SOLVE-FOR/IGNORES
JPR(4,4)	2118	ACTUAL 2ND MOMENT MATRIX, MEASUREMENT RESIDUAL
TXW(6,15)	2198	STM PARTITION ASSOCIATED WITH IGNORE PARAMETERS
AN(4,15)	2468	OBSERVATION MATRIX PARTITION ASSOCIATED WITH IGNORE PARAMETERS
GCUV(8,15)	2648	NO LONGER USED
GCUW(8,15)	2A08	NO LONGER USED
GCVW(15,15)	2DC8	ACTUAL 2ND MOMENT MATRIX, MEAS.CONS./IGNORES
GU(8,8)	34D0	NO LONGER USED
GV(15,15)	36D0	ACTUAL 2ND MOMENT MATRIX, MEAS. CONS. VECTOR
GW(15,15)	3DD8	ACTUAL 2ND MOMENT MATRIX, IGNORE PARAMETERS
GDNCN(3)	44E0	ACTUAL DYNAMIC NOISE CONSTANTS
GMNCN(15)	44F8	ACTUAL MEASUREMENT NOISE CONSTANTS
EXI(6)	4570	ACTUAL MEANS OF INITIAL STATE DEVIATIONS
EXSI(15)	45A0	ACTUAL MEANS OF INITIAL SOLVE-FOR DEVIATIONS
EU(8)	4618	NO LONGER USED
EV(15)	4658	ACTUAL MEANS OF INITIAL MEAS.COND. DEVIATIONS
EW(15)	46D0	ACTUAL MEANS OF INITIAL IGNORE DEVIATIONS
QPR(6,6)	4748	ACTUAL 2ND MOMENT MATRIX, DYNAMIC NOISE
RPR(4,4)	4868	ACTUAL 2ND MOMENT MATRIX, MEASUREMENT NOISE
GCXW(6,15)	48E8	ACTUAL 2ND MOMENT MATRIX, STATE/IGNORES
EXT(6)	4BB8	ACTUAL MEANS, UPDATED EST. ERRORS, STATE
EXST(15)	4BE8	ACTUAL MEANS, UPDATED EST. ERRORS, SOLVE-FORS
EXTP(6)	4C60	ACTUAL MEANS, PROPAGATED EST. ERRORS, STATE
EXSTP(15)	4C90	ACTUAL MEANS, PROPAGATED EST. ERRORS, SOLVE
GCXWP(6,15)	4D08	ACTUAL 2ND MOMENT MATRIX, STATE/IGNORE BEFORE PROCESSING A MEASUREMENT
GCXSWP(15,15)	4FD8	ACTUAL 2ND MOMENT MATRIX, SOLVE-FOR/IGNORE BEFORE PROCESSING A MEASUREMENT
EMRES(4)	56E0	ACTUAL MEANS, MEASUREMENT RESIDUALS
IGDNF	5700	ACTUAL DYNAMIC NOISE FLAG
IGMNF	5704	NO LONGER USED

NAME(DIM)	DISP	DEFINITION

GUI (SIZE 18C8)

PG(6,6)	0	CONTROL COVARIANCE, STATE
CXXSG(6,15)	120	CONTROL COVARIANCE, STATE/SOLVE-FOR
CXUG(6,8)	3F0	CONTROL COVARIANCE, NO LONGER USED
CXVG(6,15)	570	CONTROL COVARIANCE, STATE/MEAS. CONS.
PSG(15,15)	840	CONTROL COVARIANCE, SOLVE-FOR VECTOR
CXSUG(15,8)	F48	CONTROL COVARIANCE, NO LONGER USED
CXSVG(15,15)	1308	CONTROL COVARIANCE, SOLVE-FOR/MEAS. CONS.
XG(6)	1A10	STATE VECTOR AT TG
TG	1A40	TIME OF LAST GUIDANCE EVENT
EM(2,6)	1A48	NO LONGER USED
PHIG(6,6)	1AA8	STM FROM INITIAL TIME ON FILE TO TG

LAGS12 (SIZE 4)

K	0	CURRENT ACCELERATION POINT

LAGS15 (SIZE 4)

IELVN	0	ACCUMULATED ACCELERATION POINTS TO BE WRITTEN

LINK11 (SIZE 10)

H	0	SIGNED STEPSIZE (SEC)
T	8	TIME UP TO WHICH INTEGRATION HAS PROGRESSED

LINK38 (SIZE 3F0)

SX1(3)	0	1ST SUM FOR EQNS OF MOTION
SX2(3)	18	2ND SUM FOR EQNS OF MOTION
SV1(3,20)	30	1ST SUM FOR VARIATIONAL EQNS
SV2(3,20)	210	2ND SUM FOR VARIATIONAL EQNS

NAME(DIM)	DISP	DEFINITION

LINK39 (SIZE 10)

YMDIC	0	YEAR, MONTH, DAY IN CODE (INITIAL FILE TIME)
HMSIC	8	HOUR, MINUTE, SECOND IN CODE (SAME TIME)

MATRIX (SIZE 1C8)

A(3,3)	0	CONVERSION, SELENOCENTRIC TO SELENOGRAPHIC
ADOT(3,3)	48	DERIVATIVE OF A
B(3,3)	90	CONVERSION, EARTH INERTIAL TO EARTH BODY FIXED
C(3,3)	D8	CONVERSION, MEAN 1950 TO TRUE OF DATE
GHA	120	GREENWICH HOUR ANGLE
DXQ(18)	128	NOT USED
XP	188	X POLAR MOTION ANGLE
YP	1C0	Y POLAR MOTION ANGLE

MEAS (SIZE 2EE8)

TMN(1000)	0	SCHEDULED TIMES OF MEASUREMENTS
MCODE(1000)	1F40	CORRESPONDING TYPES OF MEASUREMENTS SCHEDULED
NMN	2EE0	NUMBER OF MEASUREMENTS SCHEDULED
MCNTR	2EE4	NUMBER OF MEASUREMENT TO BE PROCESSED NEXT

MEQEC (SIZE 48)

EQEC(3,3)	0	EQUATORIAL TO ECLIPTIC TRANSFORMATION MATRIX

MISC (SIZE 8C)

ACC	0	NO LONGER USED
FACP	8	NO LONGER USED
FACV	10	NO LONGER USED
BIA(12)	18	NO LONGER USED
IDNF	78	FLAG FOR ASSUMED DYNAMIC NOISE
ICOOR	7C	NO LONGER USED
ITR	80	NO LONGER USED
IMNF	84	NO LONGER USED
ISP2	88	NO LONGER USED

NAME(DIM)	DISP	DEFINITION

NAME (SIZE 268)

EVNM(11)	0	EVENT NAMES
MNNAME(12,3)	58	MEASUREMENT NAMES
CMPNM(30)	178	NO LONGER USED

NOVENT (SIZE 8)

NEVENT	0	NUMBER OF NEXT EVENT
II	4	NO LONGER USED

OVERPR (SIZE 10)

MMCODE	0	NEXT MEASUREMENT TYPE
NR	4	NUMBER OF ROWS IN OBSERVATION MATRIX
TRTM2	8	TIME OF NEXT MEASUREMENT OR EVENT

OVERZ (SIZE 198)

RI(6)	0	TEMPORARY STORAGE FOR STATE VECTOR.
RF(6)	30	TEMPORARY STORAGE FOR STATE VECTOR
ADA(3,6)	60	VARIATION MATRIX
TINJ	F0	NO LONGER USED
TEVN	F8	TIME OF NEXT EVENT
GA(3,6)	100	GUIDANCE MATRIX
IGP	190	GUIDANCE POLICY CODE
NOGEN	194	NO LONGER USED

PHISAV (SIZE 848)

TOLD	0	T1 FOR PHIOLD
PHIOLD(6,20)	8	STM FROM INITIAL FILE TIME TO T1
PHINEW(6,20)	3C8	STM FROM INITIAL FILE TIME TO T2 (NEW TIME)
FISAVE(6,20)	788	TEMPORARY STORAGE FOR STM

NAME(DIM)	DISP	DEFINITION
PRT (SIZE 58)		
PLANET(11)	0	NAMES OF PLANETS
PSAVE (SIZE 1A10)		
PSAV (834)	0	STORAGE FOR COVARIANCES DURING PREDICTION AND GUIDANCE EVENTS
PUNK (SIZE 10)		
IPUN	0	PUNCH FLAG TO CORREL
IPUNE	4	PUNCH FLAG FOR EIGENGECTOR EVENTS
IPUNP	8	PUNCH FLAG FOR PREDICTION EVENTS
IPUNG	C	PUNCH FLAG FOR GUIDANCE EVENTS
RLINK4 (SIZE 130)		
PVINT(6)	0	PLANET STATE AT GIVEN TIME
AEINT(6)	30	NOT USED THIS PROGRAM
SPINT(6)	60	NOT USED THIS PROGRAM
OBLINT(14)	90	NOT USED THIS PROGRAM
DUMY(6)	100	NOT USED THIS PROGRAM

NAME(DIM)	DISP	DEFINITION

RLINK5 (SIZE 528D)

Q(3)	0	S/C POSITION
QD(3)	18	S/C VELOCITY
ZDD(40,3)	30	ARRAY OF S/C ACCELERATION VECTORS
XV(3,20)	3F0	S/C POSITION PARTIALS
XVD(3,20)	5D0	S/C VELOCITY PARTIALS

RLINK9 (SIZE 60)

TWOPI	0	DOUBLE PRECISION VALUE OF 2*PI
GM(11)	8	

SCALE (SIZE 8)

SKALE	0	FACTOR USED TO SUBTRACT FRACTION OF KNOWLEDGE COVARIANCE FROM CONTROL COVARIANCE IN GUIDM

SLPOPT (SIZE 38)

DJ	0	EVAL INTERNAL
IDAY	8	DAY OF 1ST RECORD ON EPHEMERIS FILE
IYEAR	C	EVAL INTERNAL
ISPAN	10	EVAL INTERNAL
NBEPM(3)	14	BODIES FOR POLYNOMIAL COEFFICIENTS
NDEGRE(3)	20	DEGREE OF POLYNOMIALS
NCFDAY	2C	NO. OF DAYS PER CURVE FIT
ISLP50	30	EVAL INTERNAL
NBSLP	34	EVAL INTERNAL

SLPREC (SIZE D7C)

TSEC	0	SECONDS FROM START OF THIS YEAR TO MDPT OF THIS RECORD TIME INTERVAL
PPOLY(3,20,2)	8	POSITION POLYNOMIAL COEFFICIENTS
VPOLY(3,20,2)	3C8	VELOCITY POLYNOMIAL COEFFICIENTS
APOLY(3,3,10)	788	POLYNOMIAL COEFFICIENTS FOR MATRIX A
CPOLY(3,3,10	A58	POLYNOMIAL COEFFICIENTS FOR MATRIX C
PDELH(10)	D28	POLYNOMIAL COEFFICIENTS FOR DELTA H
IDAY	D78	INITIAL DAY OF THIS RECORD

NAME(DIM)	DISP	DEFINITION
STM (SIZE 4CD8)		
P(6,6)	0	COVARIANCE MATRIX, STATE VARIABLES
CXXS(6,15)	120	COVARIANCE MATRIX, STATE/SOLVE-FOR
CXU(6,8)	3F0	COVARIANCE MATRIX, NO LONGER USED
CXV(6,15)	570	COVARIANCE MATRIX, STATE/MEAS. CONS.
PS(15,15)	840	CONVAPIANCE MATRIX, SOLVE-FOR PARAMETERS
CXSU(15,8)	F48	COVARIANCE MATRIX, NO LONGER USED
CXSV(15,15)	1308	COVARIANCE MATRIX, SOLVE-FOR/MEAS. CONS.
U0(8,8)	1A10	COVARIANCE MATRIX, CONTROL VARIABLES (BURN)
V0(15,15)	1C10	COVARIANCE MATRIX, MEASUREMENT CONSIDERS
PHI(6,6)	2318	STATE-TO-STATE TRANSITION MATRIX
TXXS(6,15)	2438	SOLVE-FOR-TO-STATE TRANSITION MATRIX
TXU(6,8)	2708	CONTROL-TO-STATE TRANSITION MATRIX
Q(6,6)	2888	DYNAMIC NOISE COVARIANCE MATRIX
R(4,4)	29A8	MEASUREMENT NOISE OBSERVATION MATRIX
AK(6,4)	2A28	FILTER GAIN MATRIX, STATE PARTITION
S(15,4)	2AE8	FILTER GAIN MATRIX, SOLVE-FOR PARTITION
H(4,6)	2CC8	OBSERVATION MATRIX PARTITION, STATE
AM(4,15)	2D88	OBSERVATION MATRIX PARTITION, SOLVE-FOR
G(4,8)	2F68	OBSERVATION MATRIX PARTITION, NOT USED
AL(4,15)	3068	OBSERVATION MATRIX PARTITION, MEAS. CONS.
HPHR	3248	STORAGE FOR MATRIX AJ WHILE COMPUTING AK
PP(6,6)	32C8	COV. MATRIX, STATE, JUST BEFORE MEAS.
CXXSP(6,15)	33E8	COV. MAT., STATE/SOLVE, BEFORE MEAS.
CXUP(6,8)	3688	COV. MAT., NO LONGER USED
CXVP(6,15)	3838	COV. MAT., STATE/MEAS. CONS. BEFORE MEAS.
PSP(15,15)	3B08	COV. MAT., SOLVE-FOR, BEFORE MEAS.
CXSUP(15,15)	4210	COV. MAT., NO LONGER USED
CXSVP(15,15)	45D0	COV. MAT., SOLVE-FOR/MEAS. CONS. BEFORE MEAS.

STVEC (SIZE 1B8)		
XI(6)	0	STATE VECTOR AT TRTM1
XF(6)	30	STATE VECTOR AT TRTM2
XB(6)	60	NO LONGER USED
NDIM1	90	DIMENSION OF SOLVE-FOR VECTOR
NDIM2	94	=0
NDIM3	98	DIMENSION OF MEAS. CONS. VECTOR
IAUGIN(24)	9C	INPUT ARRAY OF AUGMENTATION CODES

TIM (SIZE 30)		
DATEJ	0	JULIAN DATE (REFERENCED TO 1950) OF TRTMB
TRTM1	8	TIME IN DAYS (SINCE ZERO) OF LAST
DELTM	10	TIME INTERVAL FOR PROPAGATION
FNTM	18	FINAL TIME
UNIVT	20	UNIVERSAL TIME

NAME(DIM)	DISP	DEFINITION
VARMAT (SIZE AC)		
REXV(3)	0	INPUT IMPULSIVE INSERTION DELTA-V VECTOR
RADA(3,6)	18	INPUT VARIATION MATRIX
IFVMRI	A8	=0, RADA NOT INPUT =1, RADA INPUT

NAME(DIM)	DISP	DEFINITION
VM (SIZE 1D4)		
ALNGTH	0	UNITS/A.U. (VALUE SET FOR KM)
TM	8	UNITS/DAY (VALUE SET FOR SEC)
DELTP	10	NO LONGER USED
RC(6)	18	NO LONGER USED
DC	48	NO LONGER USED
RSI(3)	50	NO LONGER USED
VSI(3)	68	NO LONGER USED
DSI	80	NO LONGER USED
RVS(6)	88	NO LONGER USED
VMU	B8	NO LONGER USED
B	C0	NO LONGER USED
BDT	C8	NO LONGER USED
BDR	D0	NO LONGER USED
DELTH	D8	NO LONGER USED
TIMINT	E0	NO LONGER USED
RE(6)	E8	NO LONGER USED
RTP(6)	118	NO LONGER USED
CAINC	148	NO LONGER USED
RCA	150	NO LONGER USED
TACA	158	NO LONGER USED
SSS(3)	160	NO LONGER USED
NLP	178	LAUNCH PLANET ID (EARTH)
NEP	17C	EPHEMERIS PLANET ID (EARTH)
NBOD	180	NO LONGER USED
NB	184	NO LONGER USED
NTP	1B0	TARGET PLANET ID (EARTH)
INPR	1B4	NO LONGER USED
IPROB	188	PROBLEM NUMBER (INPUT)
ISPH	1B6	NO LONGER USED
INCMT	1C0	NO LONGER USED
IEPHEM	1C4	NO LONGER USED
ICL	1C8	NO LONGER USED
IPRINT	1CC	EACH IPRINT-TH MEASUREMENT IS OUTPUT
ICL2	1C0	NO LONGER USED

NAME(DIM)	DISP	DEFINITION
XXXL (SIZE 224)		
XSL(15)	0	SOLVE-FOR PARAMETER NAMES
XU(8)	78	NO LONGER USED
XV(15)	88	MEASUREMENT CONSIDER PARAMETER NAMES
XLAB(6)	130	STATE VECTOR COMPONENT NAMES
XNM(24)	160	AUGMENTATION PARAMETER LABELS
KPRINT	220	=0, PRINT ONLY PHI*P*PHI(T) =1, PRINT ALL COVARIANCE DATA

3.3.2 COMMON VARIABLES IN ALPHABETICAL ORDER

VARIABLE(DIM)	BLOCK	DEFINITION
A(3,3)	MATRIX	CONVERSION, SELENOCENTRIC TO SELENOGRAPHIC
ACC	MISC	NO LONGER USED
ADA(3,6)	OVERZ	VARIATION MATRIX
ADOT(3,3)	MATRIX	DERIVATIVE OF A
AEINT(6)	RLINK4	NOT USED THIS PROGRAM
AINC7	EVENT	NO LONGER USED
AK(6,4)	STM	FILTER GAIN MATRIX, STATE PARTITION
AL(4,15)	STM	OBSERVATION MATRIX PARTITION, MEAS. CONS.
ALNGTH	VM	UNITS/A.U. (VALUE SET FOR KM)
AM(4,15)	STM	OBSERVATION MATRIX PARTITION, SOLVE-FOR
ANODE7	EVENT	NO LONGER USED
AN(4,15)	GENRL	OBSERVATION MATRIX PARTITION ASSOCIATED WITH IGNORE PARAMETERS
APOLY(3,3,10)	SLPREC	POLYNOMIAL COEFFICIENTS FOR MATRIX A
B(3,3)	MATRIX	CONVERSION, EARTH INERTIAL TO EARTH BODY FIXED
B	VM	NO LONGER USED
BDR	VM	NO LONGER USED
BDT	VM	NO LONGER USED
BIA(12)	MISC	NO LONGER USED
BRNTIM	EVENT	DURATION (DAYS) OF FINAL INSERTION BURN
C(3,3)	MATRIX	CONVERSION, MEAN 1950 TO TRUE OF DATE
CAINC	VM	NO LONGER USED
CECC(4,9)	BLK	NO LONGER USED
CINC(4,9)	BLK	NO LONGER USED
CMEAN(4,9)	BLK	NO LONGER USED
CMIN(6,15)	GAINC	CORRELATION MATRIX, POS/VEL AND SOLVE-FORS
CMPNM(30)	NAME	NO LONGER USED
COMEG(4,9)	BLK	NO LONGER USED
COMEGT(4,9)	BLK	NO LONGER USED
CPLU(6,15)	GAINC	CORRELATION MATRIX, POS/VEL AND SOLVE-FORS
CPOLY(3,3,10)	SLPREC	POLYNOMIAL COEFFICIENTS FOR MATRIX C
CSAX(2,9)	BLK	NO LONGER USED
CXSU(15,8)	STM	COVARIANCE MATRIX, NO LONGER USED
CXSUG(15,8)	GUI	CONTROL COVARIANCE, NO LONGER USED
CXSUP(15,15)	STM	COV. MAT., NO LONGER USED
CXSV(15,15)	STM	COVARIANCE MATRIX, SOLVE-FOR/MEAS. CONS.
CXSVG(15,15)	GUI	CONTROL COVARIANCE, SOLVE-FOR/MEAS. CONS.
CXSVP(15,15)	STM	COV. MAT., SOLVE-FOR/MEAS. CONS. BEFORE MEAS.
CXU(6,8)	STM	COVARIANCE MATRIX, NO LONGER USED
CXUG(6,8)	GUI	CONTROL COVARIANCE, NO LONGER USED
CXUP(6,8)	STM	COV. MAT., NO LONGER USED
CXV(6,15)	STM	COVARIANCE MATRIX, STATE/MEAS. CONS.
CXVG(6,15)	GUI	CONTROL COVARIANCE, STATE/MEAS. CONS.
CXVP(6,15)	STM	COV. MAT., STATE/MEAS. CONS. BEFORE MEAS.
CXXS(6,15)	STM	COVARIANCE MATRIXT ST 0 0
CXXSG(6,15)	GUI	CONTROL COVARIANCE, STATE/SOLVE-FOR
CXXSP(6,15)	STM	COV. MAT., STATE/SOLVE, BEFORE MEAS.

VARIABLE(DIM)	BLOCK	DEFINITION
DATEJ	TIM	JULIAN DATE (REFERENCED TO 1950) OF TRTMB
DC	VM	NO LONGER USED
DELTH	VM	NO LONGER USED
DELTM	TIM	TIME INTERVAL FOR PROPAGATION
DELTP	VM	NO LONGER USED
DJ	SLPOPT	EVAL INTERNAL
DNCN(3)	CONST	DYNAMIC NOISE CONSTANTS
DSI	VM	NO LONGER USED
DUMY(6)	RLINK4	NOT USED THIS PROGRAM
DV8(3)	EVENT	NO LONGER USED
DXQ(18)	MATRIX	NOT USED
ECC7	EVENT	NO LONGER USED
EE(4)	GENGD	ACTUAL MEANS OF EXECUTION ERROR PARAMETERS
EEE(4)	GENGD	VARIANCES OF EXECUTION ERROR PARAMETERS
EM(2,6)	GUI	NO LONGER USED
EMN(15)	BLK	NO LONGER USED
EMRES(4)	GENRL	ACTUAL MEANS, MEASUREMENT RESIDUALS
EM1	DPNUM	DOUBLE-PRECISION VALUE OF 1.E-1
EM2	DPNUM	DOUBLE-PRECISION VALUE OF 1.E-2
EM3	DPNUM	DOUBLE-PRECISION VALUE OF 1.E-3
EM4	DPNUM	DOUBLE-PRECISION VALUE OF 1.E-4
EM5	DPNUM	DOUBLE-PRECISION VALUE OF 1.E-5
EM6	DPNUM	DOUBLE-PRECISION VALUE OF 1.E-6
EM7	DPNUM	DOUBLE-PRECISION VALUE OF 1.E-7
EM8	DPNUM	DOUBLE-PRECISION VALUE OF 1.E-8
EM9	DPNUM	DOUBLE-PRECISION VALUE OF 1.E-9
EM13	DPNUM	DOUBLE-PRECISION VALUE OF 1.E-13
EPS	CONST	EARTH OBLIQUITY
EP50	DPNUM	DOUBLE-PRECISION VALUE OF 1.E+50
EQEC(3,3)	MEQEC	EQUATORIAL TO ECLIPTIC TRANSFORMATION MATRIX
EU(8)	GENRL	NO LONGER USED
EV(15)	GENRL	ACTUAL MEANS OF INITIAL MEAS.COND. DEVIATIONS
EVNM(11)	NAME	EVENT NAMES
EW(15)	GENRL	ACTUAL MEANS OF INITIAL IGNORE DEVIATIONS
EXI(6)	GENRL	ACTUAL MEANS OF INITIAL STATE DEVIATIONS
EXSI(15)	GENRL	ACTUAL MEANS OF INITIAL SOLVE-FOR DEVIATIONS
EXST(15)	GENRL	ACTUAL MEANS, UPDATED EST. ERRORS, SOLVE-FORS
EXSTP(15)	GENRL	ACTUAL MEANS, PROPAGATED EST. ERRORS, SOLVE
EXT(6)	GENRL	ACTUAL MEANS, UPDATED EST. ERRORS, STATE
EXTP(6)	GENRL	ACTUAL MEANS, PROPAGATED EST. ERRORS, STATE
FACP	MISC	NO LONGER USED
FACV	MISC	NO LONGER USED
FISAVE(6,20)	PHISAV	TEMPORARY STORAGE FOR STM
FNTM	TIM	FINAL TIME
FOP	CONST2	OFF-DIAGONAL ANNIHILATION VALUE (POSITION)
FOV	CONST2	OFF-DIAGONAL ANNIHILATION VALUE (VELOCITY)
G(4,8)	STM	OBSERVATION MATRIX PARTITION, NOT USED
GA(3,6)	OVERZ	GUIDANCE MATRIX

VARIABLE(DIM)	BLOCK	DEFINITION
		ACTUAL CONTROL SECOND MOMENT MATRICES
GCXSUG(15,8)	GENGD1	NO LONGER USED
GCXSVG(15,15)	GENGD1	SOLVE-FOR/MEASUREMENT CONSIDERS
GCXSWG(15,15)	GENGD1	SOLVE-FOR/IGNORE PARAMETERS
GCXUG(6,8)	GENGD1	NO LONGER USED
GCXVG(6,15)	GENGD1	STATE/MEASUREMENT CONSIDERS
GCXWG(6,15)	GENGD1	STATE/IGNORE PARAMETERS
GCXXSG(6,15)	GENGD1	STATE/SOLVE-FOR VECTOR
GPG(6,6)	GENGD1	STATE
GPSG(15,15)	GENGD1	SOLVE-FOR VECTOR
GCUV(8,15)	GENRL	NO LONGER USED
GCUW(8,15)	GENRL	NO LONGER USED
GCVW(15,15)	GENRL	ACTUAL 2ND MOMENT MATRIX, MEAS.CONS./IGNORES
GCXSU(15,8)	GENRL	NO LONGER USED
GCXSV(15,15)	GENRL	ACTUAL 2ND MOMENT MATRIX, SOLVE-FOR/MS. CN.
GCXSW(15,15)	GENRL	ACTUAL 2ND MOMENT MATRIX, SOLVE-FOR/IGNORES
GCXSWP(15,15)	GENRL	ACTUAL 2ND MOMENT MATRIX, SOLVE-FOR/IGNORE BEFORE PROCESSING A MEASUREMENT
GCXU(6,8)	GENRL	NO LONGER USED
GCXV(6,15)	GENRL	ACTUAL 2ND MOMENT MATRIX, STATE/MEAS. CONS.
GCXW(6,15)	GENRL	ACTUAL 2ND MOMENT MATRIX, STATE/IGNORES
GCXWP(6,15)	GENRL	ACTUAL 2ND MOMENT MATRIX, STATE/IGNORE BEFORE PROCESSING A MEASUREMENT
GCXXS(6,15)	GENRL	ACTUAL 2ND MOMENT MATRIX, STATE/SOLVE-FORS
GDNCN(3)	GENRL	ACTUAL DYNAMIC NOISE CONSTANTS
GHA	MATRIX	GREENWICH HOUR ANGLE
GM(11)	RLINK9	
GMNCN(15)	GENRL	ACTUAL MEASUREMENT N-ISE C-NSTANTS
GP(6,6)	GENRL	ACTUAL STATE 2ND M-MENT -+T-+0
GPS(15,15)	GENRL	ACTUAL 2ND MOMENT MATRIX, SOLVE-FOR VECTOR
GU(8,8)	GENRL	NO LONGER USED
GV(15,15)	GENRL	ACTUAL 2ND MOMENT MATRIX, MEAS. CONS. VECTOR
GW(15,15)	GENRL	ACTUAL 2ND MOMENT MATRIX, IGNORE PARAMETERS
H	LINK11	SIGNED STEPSIZE (SEC)
H(4,6)	STM	OBSERVATION MATRIX PARTITION, STATE
HALF	DPNUM	DOUBLE-PRECISION VALUE OF HALF (0.5)
HMSIC	LINK39	HOUR, MINUTE, SECOND IN CODE (SAME TIME)
HPHR	STM	STORAGE FOR MATRIX AJ WHILE COMPUTING AK
HP7	EVENT	NO LONGER USED
IAUGIN(24)	STVEC	INPUT ARRAY OF AUGMENTATION CODES
IAUGW(24)	GCA	IGNORE PARAMETER AUGMENTATION VECTOR
ICDQ3(20)	EVENT	NO LONGER USED
ICDT3(20)	EVENT	CODES FOR GUIDANCE POLICIES
ICENTB	FLAGS2	CENTRAL BODY NUMBER
ICL	VM	NO LONGER USED
ICL2	VM	NO LONGER USED
ICOOR	MISC	NO LONGER USED
IDAY	SLPOPT	DAY OF 1ST RECORD ON EPHEMERIS FILE

VARIABLE(DIM)	BLOCK	DEFINITION
IDAY	SLPREC	INITIAL DAY OF THIS RECORD
IDNF	MISC	FLAG FOR ASSUMED DYNAMIC NOISE
IEIG	EVENT	NO LONGER USED
IELVN	LAGS15	ACCUMULATED ACCELERATION POINTS TO BE WRITTEN
IEPHEM	VM U	NO LONGER USED
IEVNT(50)	EVENT	CODED EVENT TYPES CORRESPONDING TO REV TIMES
IFVMRI	VARMAT	=0, RADA NOT INPUT =1, RADA INPUT
IGDNF	GENRL	ACTUAL DYNAMIC NOISE FLAG
IGEN	GCA	=0, PERFORM NO GENERALIZED COVARIANCE ANALYSIS =1, PERFORM GENERALIZED COVARIANCE ANALYSIS
IGMNF	GENRL	NO LONGER USED
IGP	OVERZ	GUIDANCE POLICY CODE
IHYP1	EVENT	NO LONGER USED
II	NOVENT	NO LONGER USED
IIPOL	EVENT	NO LONGER USED
IMNF	MISC	NO LONGER USED
INC	COM	NO LONGER USED
INCMNT	COM	NO LONGER USED
INCMT	VM	NO LONGER USED
INCPR	COM	NO LONGER USED
IND(25)	FLAGS2	CURRENT SECTION FLAGS
INDSEC(10,3)	FLAGS2	SECTION FLAGS
INPR	VM	NO LONGER USED
IPRINT	VM	EACH IPRINT-TH MEASUREMENT IS OUTPUT
IOP7	EVENT	NO LONGER USED
IPOL	EVENT	NO LONGER USED
IPR	COM	NO LONGER USED
IPROB	VM	PROBLEM NUMBER (INPUT)
IPRT(4)	COM	NO LONGER USED
IPUN	PUNK	PUNCH FLAG TO CORREL
IPUNE	PUNK	PUNCH FLAG FOR EIGENGECTOR EVENTS
IPUNG	PUNK	PUNCH FLAG FOR GUIDANCE EVENTS
IPUNP	PUNK	PUNCH FLAG FOR PREDICTION EVENTS
ISLP50	SLPOPT	EVAL INTERNAL
ISPAN	SLPOPT	EVAL INTERNAL
ISPH	VM	NO LONGER USED
ISP2	MISC	NO LONGER USED
ITR	MISC	NO LONGER USED
ITRAT	COM	NO LONGER USED
IYEAR	SLPOPT	EVAL INTERNAL
JPR(4,4)	GENRL	ACTUAL 2ND MOMENT MATRIX, MEASUREMENT RESIDUAL
K	LAGS12	CURRENT ACCELERATION POINT
KOUNT	COM	NO LONGER USED
KPRINT	XXXL	=0, PRINT ONLY PHI*P*PHI(T) =1, PRINT ALL COVARIANCE DATA
MCNTR	MEAS	NUMBER OF MEASUREMENT TO BE PROCESSED NEXT
MCODE(1000)	MEAS	CORRESPONDING TYPES OF MEASUREMENTS SCHEDULED
MMCODE	OVERPR	NEXT MEASUREMENT TYPE

VARIABLE(DIM)	BLOCK	DEFINITION
MNCN(12)	CONST	MEASUREMENT NOISE CONSTANTS
MNNAME(12,3)	NAME	MEASUREMENT NAMES
MUPLAN(11)	BLK	GRAVITATIONAL CONSTANTS (KM((3/SEC**2)
NAE	EVENT	NO LONGER USED
NAF6(20)	EVENT	NO LONGER USED
NB	VM	NO LONGER USED
NBOD	VM	NO LONGER USED
NBSLP	SLPOPT	EVAL INTERNAL
NCFDAY	SLPOPT	NO. OF DAYS PER CURVE FIT
NDEGRE(3)	SLPOPT	DEGREE OF POLYNOMIALS
NBEPM(3)	SLPOPT	BODIES FOR POLYNOMIAL COEFFICIENTS
NBODY	COM	NO LONGER USED
NBODYI	COM	NO LONGER USED
NDIM1	STVEC	DIMENSION OF SOLVE-FOR VECTOR
NDIM2	STVEC	=0
NDIM3	STVEC	DIMENSION OF MEAS. CONS. VECTOR
NDIM4	GCA	DIMENSION OF IGNORE PARAMETER STATE
NEP	VM	EPHEMERIS PLANET ID (EARTH)
NEV	EVENT	NUMBER OF EVENTS SCHEDULED
NEVENT	NOVENT	NUMBER OF NEXT EVENT
NEV1	EVENT	NUMBER OF SCHEDULED EIGENVECTOR EVENTS
NEV2	EVENT	NUMBER OF SCHEDULED PREDICTION EVENTS
NEV3	EVENT	NUMBER OF SCHEDULED GUIDANCE EVENTS
NEV4	EVENT	NUMBER OF SCHEDULED INSERTION EVENTS
NEV5	EVENT	NO LONGER USED
NEV6	EVENT	NO LONGER USED
NEV7	EVENT	NO LONGER USED
NEV8	EVENT	NO LONGER USED
NEV9	EVENT	NO LONGER USED
NEV10	EVENT	NO LONGER USED
NEV11	EVENT	NO LONGER USED
NGE	EVENT	COUNT OF GUIDANCE EVENTS
NLP	VM	LAUNCH PLANET ID (EARTH)
NMN	MEAS	NUMBER OF MEASUREMENTS SCHEDULED
NO(11)	BLK	NO LONGER USED
NOGEN	OVERZ	NO LONGER USED
NPE	EVENT	COUNT OF PREDICTI-N EVE-TO
NQE	EVENT	NO LONGER USED
NR	OVERPR	NUMBER OF ROWS IN OBSERVATION MATRIX
NSECTN	FLAGS2	CURRENT SECTION NUMBER
NST	CONST	NUMBER OF STATIONS TO BE USED (MAXIMUM 3)
NTP	VM	TARGET PLANET ID (EARTH)
NTSEQS	FLAGS2	TOTAL NUMBER OF SECTIONS
OBLINT(14)	RLINK4	NOT USED THIS PROGRAM
OMEGA	CONST	ROTATION RATE OF EARTH
ONE	DPNUM	DOUBLE-PRECISION VALUE OF ONE (1.0)
P(6,6)	STM	COVARIANCE MATRIX. STATE VARIABLES
PDFLH(10)	SLPREC	POLYNOMIAL COEFFICIENTS FOR DELTA H
PERP7	EVENT	NO LONGER USED

VARIABLE(DIM)	BLOCK	DEFINITION
PG(6,6)	GUI	CONTROL COVARIANCE, STATE
PHI(6,6)	STM	STATE-TO-STATE TRANSITION MATRIX
PHIG(6,6)	GUI	STM FROM INITIAL TIME ON FILE TO TG
PHINEW(6,20)	PHISAV	STM FROM INITIAL FILE TIME TO T2 (NEW TIME)
PHIOLD(6,20)	PHISAV	STM FROM INITIAL FILE TIME TO T1
PI	COM	MATHEMATICAL CONSTANT PI
PLANET(11)	PRT	NAMES OF PLANETS
PMASS(11)	BLK	GRAVITATIONAL CONSTANTS (A.U.**3/DAY**2)
PMIN(6,6)	GAINC	POS/VEL COVARIANCE BEFORE MEASUREMENT (WLS)
PP(6,6)	STM	COV. MATRIX, STATE, JUST BEFORE MEAS.
PPLU(6,6)	GAINC	POS/VEL COVARIANCE AFTER MEASUREMENT (WLS)
PPOLY(3,20,2)	SLPREC	POSITION POLYNOMIAL COEFFICIENTS
PS(15,15)	STM	CONVARIANCE MATRIX, SOLVE-FOR PARAMETERS
PSAV (834)	PSAVE	STORAGE F-R COVARIANCES DURING PREDICTI-N AND GUIDANCE EVENTS
PSG(15,15)	GUI	CONTROL COVARIANCE, SOLVE-FOR VECTOR
PSMIN(15,15)	GAINC	SOLVE-FOR COVARIANCE BEFORE MEASUREMENT (WLS)
PSP(15,15)	STM	COV. MAT., SOLVE-FOR, BEFORE MEAS.
PSPLU(15,15)	GAINC	SOLVE-FOR COVARIANCE AFTER MEASUREMENT (WLS)
PVINT(6)	RLINK4	PLANET STATE AT GIVEN TIME
P7	EVENT	NO LONGER USED
Q(3)	RLINK5	S/C POSITION
Q(6,6)	STM	DYNAMIC NOISE COVARIANCE MATRIX
QD(3)	RLINK5	S/C VELOCITY
QPR(6,6)	GENRL	ACTUAL 2ND MOMENT MATRIX, DYNAMIC NOISE
R(4,4)	STM	MEASUREMENT NOISE OBSERVATION MATRIX
RAD	COM	NUMBER OF DEGREES PER RADIAN
RADA(3,6)	VARMAT	INPUT VARIATION MATRIX
RADIUS(11)	BLK	RADII OF THE PLANETS (A.U.)
RC(6)	VM	NO LONGER USED
RCA	VM	NO LONGER USED
RE(6)	VM	NO LONGER USED
REXV(3)	VARMAT	INPUT IMPULSIVE INSERTION DELTA-V VECTOR
RF(6)	OVERZ	TEMPORARY STORAGE FOR STATE VECTOR
RI(6)	OVERZ	TEMPORARY STORAGE FOR STATE VECTOR
RMASS(11)	BLK	GRAVITATIONAL CONSTANTS (RELATIVE TO SUN)
RPP(4,4)	GENRL	ACTUAL 2ND MOMENT MATRIX, MEASUREMENT NOISE
RSAVE(6)	GAINC	STATE VECTOR AT TLAST
RSI(3)	VM	NO LONGER USED
RTP(6)	VM	NO LONGER USED
RVS(6)	VM	NO LONGER USED
S(15,4)	STM	FILTER GAIN MATRIX, SOLVE-FOR PARTITION
SAL(3)	CONST	STATION ALTITUDES (ABOVE RADIUS OF EARTH)
SIGALP	EVENT	VARIANCE OF ERROR IN POINTING ANGLE 1
SIGBET	EVENT	VARIANCE OF ERROR IN POINTING ANGLE 2
SIGPRO	EVENT	VARIANCE OF PROPORTIONALITY ERROR
SIGRES	EVENT	VARIANCE OF RESOLUTION ERROR
SKALE	SCALE	FACTOR USED TO SUBTRACT FRACTION OF KNOWLEDGE COVARIANCE FROM CONTROL COVARIANCE IN GUIDM
SLAT(3)	CONST	STATION LATITUDES
SLON(3)	CONST	STATION LONGITUDES

VARIABLE(DIM)	BLOCK	DEFINITION
SMJR(2,9)	BLK	NO LONGER USED
SPHERE(11)	BLK	SPHERES OF INFLUENCE (A.U.)
SPINT(6)	RLINK4	NOT USED THIS PROGRAM
SSS(3)	VM	NO LONGER USED
SV1(3,20)	LINK38	1ST SUM FOR VARIATIONAL EQNS
SV2(3,20)	LINK38	2ND SUM FOR VARIATIONAL EQNS
SX1(3)	LINK38	1ST SUM FOR EQNS OF MOTION
SX2(3)	LINK38	2ND SUM FOR EQNS OF MOTION
T	BLK	TRAJECTORY TIME IN DAYS
T	LINK11	TIME UP TO WHICH INTEGRATION HAS PROGRESSED
TACA	VM	NO LONGER USED
TAU7	EVENT	NO LONGER USED
TEV(50)	EVENT	SCHEDULED TIMES OF EVENTS
TEVN	OVERZ	TIME OF NEXT EVENT
TG	GUI	TIME OF LAST GUIDANCE EVENT
THREE	DPNUM	DOUBLE-PRECISION VALUE OF THREE (3.0)
TIMINT	VM	NO LONGE- USED
TINJ	OVERZ	NO LONGER USED
TLAST	GAINC	TIME WHEN MEASUREMENT LAST PROCESSED
TM	VM	UNITS/DAY (VALUE SET FOR SEC)
TMN(1000)	MEAS	SCHEDULED TIMES OF MEASUREMENTS
TOLD	PHISAV	T1 FOR PHIOLD
TPT2(20)	EVENT	TIMES PREDICTED TO IN PREDICTION EVENTS
TRTM1	TIM	TIME IN DAYS (SINCE ZERO) OF LAST
TRTM2	OVERPR	TIME OF NEXT MEASUREMENT OR EVENT
TSEC	SLPREC	SECONDS FROM START OF THIS YEAR TO MDPT OF THIS RECORD TIME INTERVAL
TWO	DPNUM	DOUBLE-PRECISION VALUE OF TWO (2.0)
TWOPI	DPNUM	DOUBLE-PRECISION VALUE OF 2.*PI
TWOPI	RLINK9	DOUBLE PRECISION VALUE OF 2.*PI
TXU(6,8)	STM	CONTROL-TO-STATE TRANSITION MATRIX
TXW(6,15)	GENRL	STM PARTITION ASSOCIATED WITH IGNORE PARAMETERS
TXXS(6,15)	STM	SOLVE-FOR-TO-STATE TRANSITION MATRIX
UNIVT	TIM	UNIVERSAL TIME
UST(3)	CONST2	X-DIRECTION COSINE FOR STARS
U0(8,8)	STM	COVARIANCE MATRIX, CONTROL VARIABLES (BURN)
VMU	VM	NO LONGER USED
VPOLY(3,20,2)	SLPREC	VELOCITY POLYNOMIAL COEFFICIENTS
VST(3)	CONST2	Y-DIRECTION COSINE FOR STARS
V0(15,15)	STM	COVARIANCE MATRIX, MEASUREMENT CONSIDERS
VSI(3)	VM	NO LONGER USED
WST(3)	CONST2	Z-DIRECTION COSINE FOR STARS
XP(6)	STVEC	NO LONGER USED
XF(6)	STVEC	STATE VECTOR AT TRTM2
XG(6)	GUI	STATE VECTOR AT TG
XI(6)	STVEC	STATE VECTOR AT TRTM1
XIG(15)	GCA	IGNORE PARAMETER LABLES
XLAB(6)	XXXL	STATE VECTOR COMPONENT NAMES
XNM(24)	XXXL	AUGMENTATION PARAMETER LABELS
XP(6)	BLK	STATE VECTOR OF PLANET

VARIABLE(DIM)	BLOCK	DEFINITION
XP	MATRIX	X POLAR MOTION ANGLE
XSL(15)	XXXL	SOLVE-FOR PARAMETER NAMES
XU(8)	XXXL	NO LONGER USED
XV(3,20)	RLINK5	S/C POSITION PARTIALS
XV(15)	XXXL	MEASUREMENT CONSIDER PARAMETER NAMES
XVD(3,20)	RLINK5	S/C VELOCITY PARTIALS
YMDIC	LINK39	YEAR, MONTH, DAY IN CODE (INITIAL FILE TIME)
YP	MATRIX	Y POLAR MOTION ANGLE
ZDD(40,3)	RLINK5	ARRAY OF S/C ACCELERATION VECTORS
ZERO	DPNUM	DOUBLE-PRECISION VALUE OF ZERO (0.0)

4. INDIVIDUAL SUBROUTINE DOCUMENTATION

This chapter provides the detailed documentation of the subroutines comprising the STEAP-L programs. The subroutine hierarchy for NOMINAL and ERRAN are defined in Figures 2.1 and 3.1. An alphabetical listing of all subroutines appearing in STEAP-L is given in Table 4.1 with a description of the program(s) that use the subroutine. Subroutines that are part of the Goddard Trajectory Determination System (GTDS) and are unchanged are not documented in this report as existing documentation is available at GSFC. Table 4.2 lists the purposes of the documented subroutines (again in alphabetical order) for convenient cross-reference.

The following pages then detail each subroutine in alphabetical order. The level of documentation of the subroutines is based on their complexity. Simple utility routines are described by defining their purpose, call sequence, and input and output arguments. The documentation of more complicated routines defines local and common variables computed by the routines, mathematical analysis, and flowcharts.

TABLE 4.1 ALPHABETICAL LISTING OF SUBROUTINES

ACOSH (N)	DMADD (N)	FORCES (N-G)	MENO (E)	SLOOZO (N)
ATCEGV (E)	DMATPY (NE)	FORCV (N-G)	MOMENT (E)	STAPRL (E)
BLOCK DATA (N-G)	DMSUB (N)	GAIN1 (E)	MSTART (N-G)	STEAPE (E)
BLOCK DATA (E)	DRD (N)	GDATA (E)	NEWTAR (N-G)	STEAPN (N)
BURN (N)	DSHIFT (N)	GENGID (E)	NTM (E)	STMPR (E)
BURNV (N)	DSUECT (N)	GETCOW (E)	OBSTIM (N-G)	STVCPR (E)
CALJUL (N)	DUNIT (N)	GHA (E)	OBSTI1 (N-G)	SUMS (N-G)
CAREL (N)	DUXV (N)	GIDANS (N)	ORBEND (N)	SYMTRK (E)
CDATE (N-G)	DVCOMB (N)	GNAVM (E)	ORBINT (N)	SYMTRZ (E)
CONSEC (N-G)	DVECRD (N)	GPRINT (E)	ORBWRT (N)	TCON (N-G)
CORREL (E)	DVMAG (E)	GQCOMP (E)	PARTE (N-G)	TDIF (N-G)
COVMAT (E)	DVSDIV (N)	GUID (E)	PECEQ (NE)	TESTH (N-G)
COWELL (N)	DVSMLT (N)	GUIDM (E)	PMASS (N-G)	TIME (E)
CSTART (E)	DVSTAT (E)	HGIDNS (N)	PMASSV (N-G)	TIMCOF (N-G)
CSTEP (N-G)	DXB (E)	HLAUNCH (N)	POLAR (N-G)	TINE (E)
DABSV (N)	DYNO (E)	HPRELM (N)	PRED (E)	TRAKM (E)
DATA (E)	DZERO (N)	HTRJTY (N)	PRELIM (N)	TRJTRY (N)
DANGMD (N)	ECOMP (N)	HZERIT (N)	PRINT3 (E)	TRNSPS (N)
DANGVZ (N)	EIGHY (E)	INITLC (N)	PSIM (E)	TWOBDY (N-G)
DAVECT (N)	ELEM (N-G)	INTP (NE-G)	PSTART (N)	VARFRC (N-G)
DDOT (N)	EPHGT (NE)	JACOBI (E)	SAVMAT (E)	XCOR (N-G)
DDOTB (E)	ERRAN (E)	LAMBRT (N)	SCHED (E)	XDCOR (N-G)
DELTIM (NE-G)	EULMX (NE)	LOOP (N)	SETEVN (E)	XSUM (N-G)
DJUL (NE-G)	EVAL (NE-G)	MATIN (NE)	SET1 (N)	ZERMAT (E)
DJUL1 (N-G)	FAPX (N-G)	MEAN (E)	SHIFT (E)	

LEGEND: N - NOMNAL ROUTINE
E - ERRAN ROUTINE
G - GTDS ROUTINE (NOT DOCUMENTED)

TABLE 4.2 PURPOSES

NAME	PURPOSE
ACOSH	To Compute the Hyperbolic Arc-Cosine
ATCEGV	To Compute Eigenvalues and Eigenvectors of Actual Target Condition and Moment Matrix
BLOCK DATA ERRAN	To Load Constants into Common Locations used in Various other Parts of the Program
BURN	To Compute the Current Acceleration due to a Finite Burn
BURNV	To Compute the Finite Burn Partials
CALJUL	To Compute Julian Date from Calendar Date or Vice Versa
CAREL	To Transform Cartesian Coordinates to Conic Elements
CØRREL	To Convert Covariance Matrix Partitions to Correlation Matrix Partitions and Standard Deviations and Write them out
CØVMAT	To Convert Standard Deviation/Correlation input to Covariance Form
COWELL	To Control the Integration Logic after the Integrator has been Initialized
CSTART	To Read the Header Record on the Sequential Orbit File
DABSV	To Calculate the Magnitude of a Vector
DANGMD	To Modularize an Angle on Two PI
DANGV2	Calculate a Directed Angle Between Two Vectors in 3 space
DATA	To Read input Data, set Default Values, Initialize and Set Internal Parameters and Print Initial Conditions
DAVECT	Vector Addition
DDOT	Vector DOT Product
DDOTB	To Return the DOT (Inner) Product of Two 3-Vectors
DMADD	Matrix Addition
DMATPY	Matrix Multiplication
DMSUB	Matrix Subtraction
DRD	To Compute Latitude and Longitude of a Vector

NAME	PURPOSE
DSHIFT	To Shift one Vector into Another
DSVECT	Vector Subraction
DUNIT	To Unitize a Vector
DUXV	To Form a Vector Cross Product
DVCOMB	To Combine (add) Two Vectors, Each Multiplied by Scalars
DVECRD	To Compute a Unit Vector from its Right Ascention and Declination
DVSDIV	Vector Scalar Division
DVSMLT	Vector Scalar Multiplication
DVMAG	Calculate the Magnitude of a 3-Vector
DVSTAT	To Compute and Print Statistical Delta-V Parameters
DXB	Calculate the Cross Product of Two 3-Vectors
DYNØ	To Compute Assumed and Actual Dynamic Noise Covariance Matrix
DZERO	To Generate a Zero Vector
ECOMP	To Compute Differential Transformation Relating Target Variables to State
EIGHY	To Control the Computation of Eigenvalues and Eigenvectors
EPHGT	To Retrieve from the Direct Access SLP File, the State Vector of a Planet with Respect to the Sun at an Arbitrary Julian Date
ERRAN	To Control the Computational Flow of the Basic Cycle
EULMX	To Compute Transformation Matrices
GAIN1	To Compute the Kalman Gain Matrices
GDATA	To Initialize Generalized Covariance Quantities
GENGID	To Generate Statistics Relating to Actual Guidance Events
GETCOW	To Generate a State Vector at a Requested Time
GHA	To Compute the Greenwich-hour Angle and Universal Time
GIDANS	Dummy Link with Non-Halo Orbit Options

NAME	PURPOSE
GNAVM	To Propagate Covariances Between Measurements and Events and to Update them at Measurements
GPRINT	To Print Actual Estimation Error Statistics
GQCØMP	To Compute Actual Execution Error Statistics
GUID	To Compute the Variation, Guidance and Target Condition (BEFØRE) Covariance Matrices
GUIDM	To Control the Execution of a Guidance Event
HGIDNS	To Compute the Change Required to the Control Variables for Targeting
HLAUCH	To Compute the Injection Time
HPRELM	To Initialize Constants and Default Values, Read Input Data, and Calculate the Zero Iterate Guess
HTRJTY	To Control the Trajectory Generation Phase
HZERIT	To Compute the Initial for Targeting when IZERO = 6 or 7
INITLC	To Initialize Constants
JACØBI	To Calculate the Eigen-Values and-Vectors of a Real Symmetric Matrix
LAMBRT	To Solve Lamberts Problem for Transfers less than Two PI
LOOP	To Solve Lamberts Problem for Transfers Greater than Two PI
MATIN	To Compute the Inverse of a Matrix
MEAN	To Propagate and Update the Means of Actual State and Parameter Deviations and Estimation Errors
MENØ	To Compute Assumed and Actual Measurement Noise Covariances
MAIN	Entry Point to Program NOMNAL
MØMENT	To Convert Covariance Matrix Partitions to Correlation Matrix Partitions, Calculate Eigen-Values and -Vectors and to Print
NTM	To Read the Trajectory File and Manipulate State Transition Matrices

NAME	PURPOSE
ORBINT	To Initialize the Sequential Orbit File with Partials
ORBEND	To Write a 'Final' Record to the Sequential Orbit File
ØRBWRT	To Write Records to the Sequential Orbit File
PECEQ	To Compute the Ecliptic to Equatorial Transformation Matrix
PRELIM	Dummy Link with Non-Halo Orbit Options
PRED	To Make Prediction Event Calculations
PRINT3	To Print Measurement Information
PSIM	To Calculate the State Transition Matrix from T2 to T3 using the T1 to T2 and T1 to T3 State Transition Matrices
PSTART	To Initialize the State Partial Matrix
SAVMAT	To Store a Vector P in a Vector P1
SCHED	To Order the Measurement Schedule
SETEVN	To Control All Event Calculations
SHIFT	To Shift a Double Precision Array to Another Location
SL0020	To Minimize F(X)
SET1	To Initialize the Flags for use by Integration Routines
STAPRL	To Compute Station Location Partials
STEAPE	To Control the Error Analysis Mode of STEAP
STMPR	To Print the State Transition Matrices
STVCPR	To Print the State Vector in Several Coordinate Systems
SYMTRK	To Symmetrize a Square Matrix
SYMTRZ	To Fill the Upper-Right Triangle of a Symmetric Square Matrix whose Lower-Left Triangle was input
TIME	To Convert a Time in Seconds to Days, Hours, Minutes and Seconds
TINE	To Compute the Julian Date Relative to 1900 from the Calendar Date or Vice Versa

NAME	PURPOSE
TRAKM	To Compute the Augmented Observation Matrix Partitions
TRNSPS	To Form the Transpose of a Matrix
TRJTRY	Dummy Link with Non-Halo Orbit Options
ZERMAT	To Zero a Matrix

FUNCTION ACOSH

PURPOSE: TO COMPUTE THE HYPERBOLIC ARC-COSINE

CALLING SEQUENCE: RES=ACOSH(X)

ARGUMENTS:

X	I	VALUE OF HYPERBOLIC COSINE
ACOSH	O	HYPERBOLIC ARC-COSINE OF X

SUBROUTINES REQUIRED:
NONE

SUBROUTINE ATCEGV

PURPOSE: TO COMPUTE EIGENVALUES AND EIGENVECTORS OF ACTUAL TARGET CONDITION 2ND MOMENT MATRIX

CALLING SEQUENCE: CALL ATCEGV(III,ATC,EDT,FOV)

ARGUMENTS:

III	I	NUMBER OF ROWS IN ATC MATRIX
ATC	I	ACTUAL TARGET CONDITION MATRIX
EDT	I	ACTUAL TARGET STATE DEVIATION MEANS
FOV	I	FINAL OFF-DIAGONAL ANNIHILATION TERM FOR JACOBI

SUBROUTINES SUPPORTED: GENGID

SUBROUTINES REQUIRED: EIGHY

COMMON USED:

LOCAL SYMBOLS:

DUM1	OUTPUT MATRIX FOR JACOBI
EGVL	EIGENVALUES
PEIG	INTERMEDIATE ARRAY
ROW	INTERMEDIATE VECTOR
S	ATC COVARIANCE ARRAY(3,3)
SDUM	ATC COVARIANCE ARRAY(2,2) FOR JACOBI

BLOCK DATA - ERRAN

PURPOSE: TO LOAD CONSTANTS INTO COMMON LOCATIONS USED IN VARIOUS OTHER PARTS OF THE PROGRAM.

CALLING SEQUENCE: NONE

ARGUMENT: NONE

SUBROUTINES SUPPORTED: HALF THE SUBROUTINES USE THE CONSTANTS STORED BY THIS BLOCK DATA

SUBROUTINES REQUIRED: NONE

COMMON LOADED:

XLAB	XNM	EVNM	MNNAME	OMEGA	PI
RAD	RMASS	RADIUS	MUPLAN	PMASS	PLANET
TM	SIGHES	SIGPRO	SIGALP	SIGBET	MNCN
NLP	NEP	NTP	ALNGTH	TWOPI	DNCN
IGAIN	IGEN	EM13	EP50	IDNF	IPROB
IFVMRI	ZERO	ONE	TWO	HALF	THREE
EM1	EM2	EM3	EM4	EM4	EM5
EM6	EM7	EM8	EM9	SAL	SLAT
SLON					

The subprogram BLOCK DATA loads constants into common blocks used by the subroutines in the ERRAN program.

The arrays loaded are:

XLAB	Hollerith names of the state-vector components
XNM	Hollerith names of the augmentation parameters
EVNM	Hollerith names of the event types
MNNAME	Hollerith names of the measurement types
OMEGA	Mean sidereal rate (radians per day)
PI	π (conversion factor)
RAD	Degrees per radian (conversion factor)
RMASS	Mass ratios of the planets (mass of the sun = 1.0)
RADIUS	Planet radii in A.U.
MUPLAN	Gravitational constant times planet mass (km^3/sec^2)
PMASS	Gravitational constant times planet mass ($A.U.^3/day^2$)
PLANET	Hollerith name of the planet
TM	Number of seconds per day (conversion factor)
SIGRES SIGPRO SIGALP SIGBET	Execution error means of resolution, proportionality, and aiming angles α, β
MNCN	Noise constants for measurement types
NLP NEP NTP	Planet number for launch, ephemeris, and target planets (nominally 4, for Earth)
ALNGTH	Number of km per A.U. (conversion factor)
TWOPI	2π (conversion factor)
DNCN	Dynamic noise constants (acceleration squared)
IGAIN	Flag to choose Kallman-Schmidt filter (=1, nominal)
IGEN	Flag to use (=1) or not use (=0) generalized covariance capabilities (nominally 0)
EM13	10^{-13} (constant)
EP50	10^{+50} (constant, "infinity")
IDNF	Flag to use dynamic noise constants
IPROB	Problem number
IFVMRI	Flag set if variation matrix (η) read in (nominally 0)
ZERO ONE TWO HALF THREE	Double precision constants 0., 1., 2., .5, 3.
EM1 - EM9	Double precision constants (10^{-1}, 10^{-2}, . . ., 10^{-9})
SAL SLAT SLON	Station location constants (altitudes in kilometers above mean radius of Earth, latitudes and longitudes in degrees)

```
SUBROUTINE BURN

PURPOSE:  TO COMPUTE THE CURRENT ACCELERATION DUE TO A FINITE BURN

CALLING SEQUENCE:  CALL BURN(ACTH)

ARGUMENTS:
         ACTH     O ACCERATION VECTOR

LOCAL SYMBOLS:
         DM3
         GRAV        GRAVITATIONAL ACCELERATION
         XMASS       INITIAL SPACE CRAFT MASS

SUBROUTINES REQUIRED:
         NONE

COMMON USED:
         H           TBURN
         T           ALPHA
         SCMASS      BETA
         THRMAG      RPD
         XISP

COMMON COMPUTED:
         DMASS       COSB
         CURMAS      SINB
         COSA
         SINA
```

SUBROUTINE BURNV

PURPOSE: TO COMPUTE THE FINITE BURN PARTIALS

CALLING SEQUENCE: CALL BURNV

ARGUMENTS:
NONE

LOCAL SYMBOLS:
NONE

SUBROUTINES REQUIRED:
NONE

COMMON USED:

THRMAG	SINB
SINA	COSB
COSA	CURMAS

COMMON COMPUTED:

ACCPAR

SUBROUTINE CALJUL

PURPOSE: TO COMPUTE JULIAN DATE FROM CALENDAR DATE OR VICE VERSA

CALLING SEQUENCE: CALL CALJUL(DJ,IY,MO,ID,IH,MI,S,ICODE)

ARGUMENTS:

DJ		JULIAN DATE
IY		CALENDAR YEAR
MO		MONTH OF YEAR
ID		DAY OF MONTH
IH		HOUR OF DAY
MI		MINUTE OF HOUR
S		SECONDS OF MINUTE
ICODE	I	OPTION FLAG 0=CONVERT FROM CALENDAR DATE TO JULIAN DATE 1=CONVERT FROM JULIAN DATE TO CALENDAR DATE

SUBROUTINES REQUIRED:
NONE

SUBROUTINE CAREL

PURPOSE: TRANSFORM CARTESIAN COORDINATES TO CONIC ELEMENTS

CALLING SEQUENCE: CALL CAREL(GM,R,V,TFP,A,E,W,XI,XN,TA,PP,QQ,WW)

ARGUMENT:

ARGUMENT:	GM	I	GRAVITATIONAL CONSTANT OF THE CENTRAL BODY
	R(3)	I	POSITION VECTOR RELATIVE TO CENTRAL BODY
	V(3)	I	VELOCITY VECTOR RELATIVE TO CENTRAL BODY
	TFP	O	TIME OF FLIGHT FROM PERIAPSIS ON THE CONIC
	A	O	SEMI-MAJOR AXIS OF THE CONIC
	E	O	ECCENTRICITY OF THE CONIC
	W	O	ARGUMENT OF PERIAPSIS OF THE CONIC
	XI	O	INCLINATION OF THE CONIC TO THE REFERENCE FRAME
	XN	O	LONGITUDE OF THE ASCENDING NODE OF THE CONIC
	TA	O	INSTANTANEOUS TRUE ANOMALY OF THE CONIC
	PP(3)	O	UNIT VECTOR TOWARD PERIAPSIS ON CONIC
	QQ(3)	O	UNIT VECTOR NORMAL TO PP IN ORBITAL PLANE
	WW(3)	O	UNIT VECTOR NORMAL TO ORBITAL PLANE

SUBROUTINES REQUIRED: NONE

LOCAL SYMBOLS:	AUXF	ECCENTRIC ANOMALY (HYPERBOLIC CASE)
	AVA	MEAN ANOMALY (ELLIPTIC CASE)
	COSEA	COSINE OF THE ECCENTRIC ANOMALY (ELLIPTIC CASE)
	CTA	COSINE OF THE TRUE ANOMALY
	C	MAGNITUDE OF THE ANGULAR MOMENTUM
	DIV	INTERMEDIATE VARIABLE IN CALCULATION OF ECCENTRIC ANOMALY

EA	ECCENTRIC ANOMALY (ELLIPTIC CASE)
P	SEMI-LATUS RECTUM OF THE CONIC
RAD	DEGREES TO RADIANS CONVERSION CONSTANT
RD	TIME DERIVATIVE OF RADIUS
RM	MAGNITUDE OF CARTESIAN POSITION VECTOR
SINEA	SINE OF THE ECCENTRIC ANOMALY (ELLIPTIC CASE)
SINHF	HYPERBOLIC SINE OF AUXF
STA	SINE OF THE TRUE ANOMALY
TANG	INTERMEDIATE VARIABLE USED TO CALCULATE SINHF
VM	MAGNITUDE OF THE CARTESIAN VELOCITY VECTOR
Z	INTERMEDIATE VECTOR USED TO CALCULATE PP, QQ VECTORS

CAREL Analysis

CAREL converts the cartesian state (position and velocity) of a massless point referenced to a gravitational body to the equivalent conic elements about that body.

Let the cartesian state be denoted $\vec{r}, \vec{v}$ and let the gravitational constant of the central body be μ.

The angular momentum constant c is

$$c = |\vec{r} \times \vec{v}| \tag{1}$$

The unit normal $\hat{W}$ to the orbital plane is

$$\hat{W} = \frac{\vec{r} \times \vec{v}}{c} \tag{2}$$

The semilatus rectum p is

$$p = \frac{c^2}{\mu} \tag{3}$$

The semi-major axis a is

$$a = \frac{r}{2 - \frac{rv^2}{\mu}} \tag{4}$$

Thus $a > 0$ for elliptical motion, $a < 0$ for hyperbolic motion. The eccentricity e is

$$e = \sqrt{1 - \frac{p}{a}} \tag{5}$$

Thus $e < 1$ for elliptical motion, $e > 1$ for hyperbolic motion. The inclination of the orbit i is computed from

$$\cos i = \hat{W}_z \tag{6}$$

The longitude of the ascending node Ω is defined by

$$\tan \Omega = \frac{\hat{W}_x}{-\hat{W}_y} \tag{7}$$

The true anomaly f at the given state is computed from

$$\cos f = \frac{p - r}{e\, r} \qquad \sin f = \frac{c\dot{r}}{\mu e} \tag{8}$$

Now define an auxiliary vector $\hat{z}$ by

$$\hat{z} = \frac{r}{c}\,\vec{v} - \frac{\dot{r}}{c}\,\vec{r} \tag{9}$$

Then $\hat{P}$, the unit vector to periapsis, and $\hat{Q}$, the in-plane normal to $\hat{P}$, are defined by

$$\hat{P} = \hat{r}\cos f - \hat{z}\sin f \tag{10}$$

$$\hat{Q} = \hat{r}\sin f + \hat{z}\cos f \tag{11}$$

where $\hat{r} = \frac{\vec{r}}{r}$. The argument of periapsis ω is then computed from

$$\tan \omega = \frac{\hat{P}_z}{\hat{Q}_z} \tag{12}$$

The conic time from periapsis t_p is computed from different formulae depending upon the sign of the semi-major axis. For $a > 0$ (elliptical motion)

$$t_p = \sqrt{\frac{a^3}{\mu}}\,(E - e\sin E)$$

$$\cos E = \frac{e + \cos f}{1 + e\cos f} \qquad \sin E = \frac{\sqrt{1 - e^2}\,\sin f}{1 + e\cos f} \tag{13}$$

For $a < 0$ (hyperbolic motion) the time from periapsis is

$$t_p = \sqrt{\frac{a^3}{\mu}}\,(e\sinh H - H)$$

$$\tanh\frac{H}{2} = \sqrt{\frac{e - 1}{e + 1}}\,\tan\frac{f}{2} \tag{14}$$

Reference: Battin, R. H., Astronautical Guidance, McGraw-Hill Book Co., New York, 1964.

SUBROUTINE CORREL

PURPOSE: CONVERT COVARIANCE MATRIX PARTITIONS TO CORRELATION MATRIX PARTITIONS AND STANDARD DEVIATIONS AND WRITE THEM OUT

CALLING SEQUENCE: CALL CORREL(PP,CXXSP,PSP,CXUP,U0,CXVP,V0,CXSUP,CXSVP)

ARGUMENT:

PP	I	POSITION/VELOCITY COVARIANCE MATRIX
CXXSP	I	CORRELATION BETWEEN SOLVE-FOR PARAMETERS AND POSITION/VELOCITY STATE
PSP	I	SOLVE-FOR PARAMETER COVARIANCE MATRIX
CXUP	I	CORRELATION BETWEEN POSITION/VELOCITY STATE AND DYNAMIC CONSIDER PARAMETERS
U0	I	DYNAMIC CONSIDER PARAMETER COVARIANCE MATRIX
CXVP	I	CORRELATION BETWEEN POSITION/VELOCITY STATE AND MEASUREMENT CONSIDER PARAMETERS
V0	I	MEASUREMENT CONSIDER PARAMETER COVARIANCE MATRIX
CXSUP	I	CORRELATION BETWEEN SOLVE-FOR PARAMETERS AND DYNAMIC CONSIDER PARAMETERS
CXSVP	I	CORRELATION BETWEEN SOLVE-FOR PARAMETERS AND MEASUREMENT CONSIDER PARAMETERS

SUBROUTINES SUPPORTED:

PRINT3 SETEVN GUIDM PRED

LOCAL SYMBOLS:

DUM	INVERSE OF SQUARE ROOT OF DIAGONAL ELEMENTS IN DYNAMIC AND MEASUREMENT CONSIDER COVARIANCE PARTITIONS
IEND	COUNTER INDICATING TOTAL NUMBER OF AUGMENTED STATE VARIABLES
ROW	INTERMEDIATE COMPUTATION AND OUTPUT VECTOR
SQP	INVERSE OF THE SQUARE ROOT OF DIAGONAL ELEMENTS IN VEHICLE AND SOLVE-FOR COVARIANCE PARTITIONS
ZZ	STANDARD DEVIATION

COMMON USED:

IPUN KPRINT NDIM1 NDIM2 NDIM3 ONE
XLAB XSL XU XV

SUBROUTINE COVMAT

PURPOSE: TO CONVERT THE STANDARD DEVIATIONS AND CORRELATION INPUT TO COVARIANCES IN A SQUARE MATRIX

CALLING SEQUENCE: CALL COVMAT(P,NR,NACTUL)

ARGUMENTS:

	P	ARRAY TO BE CONVERTED
	NR	NUMBER OF ROWS USED
	NACTUL	ACTUAL DIMENSION OF MATRIX

LOCAL SYMBOLS:

	K1	INDEX
	JP	INDEX

SUBROUTINES REQUIRED: NONE

COMMON USED: NONE

SUBROUTINE COWELL

PURPOSE: TO CONTROL THE INTEGRATION LOGIC AFTER THE INTEGRATOR HAS BEEN INITIALIZED

CALLING SEQUENCE: CALL COWELL(TTO,XTO,VTO)

ARGUMENTS:

TTO	I	TIME (SEC) FROM EPOCH AT WHICH TRAJECTORY DATA IS REQUIRE
XTO	O	6 ELEMENT STATE VECTOR AT TIME TTO
VTO	O	6 BY 20 ELEMENT MATRIX OF STATE PARTIALS

LOCAL SYMBOLS:

TINTEG	TIME THAT HAS BEEN INTEGRATED TO
STIME	TIME AT INTERPOLATION
STREGT	TIME ASSOCIATED WITH LAST ACCELERATION VECTOR
IV	FLAG FOR INTP TO INDICATE POSITION AND VELOCITY ARE REQUIRED
ISW	FLAG FROM TESTH TO INDICATE IF STEP SIZE HAS CHANGED
IERR	FLAG FROM CSTEP TO INDICATE THAT INTEGRATION STEP HAS FAILED

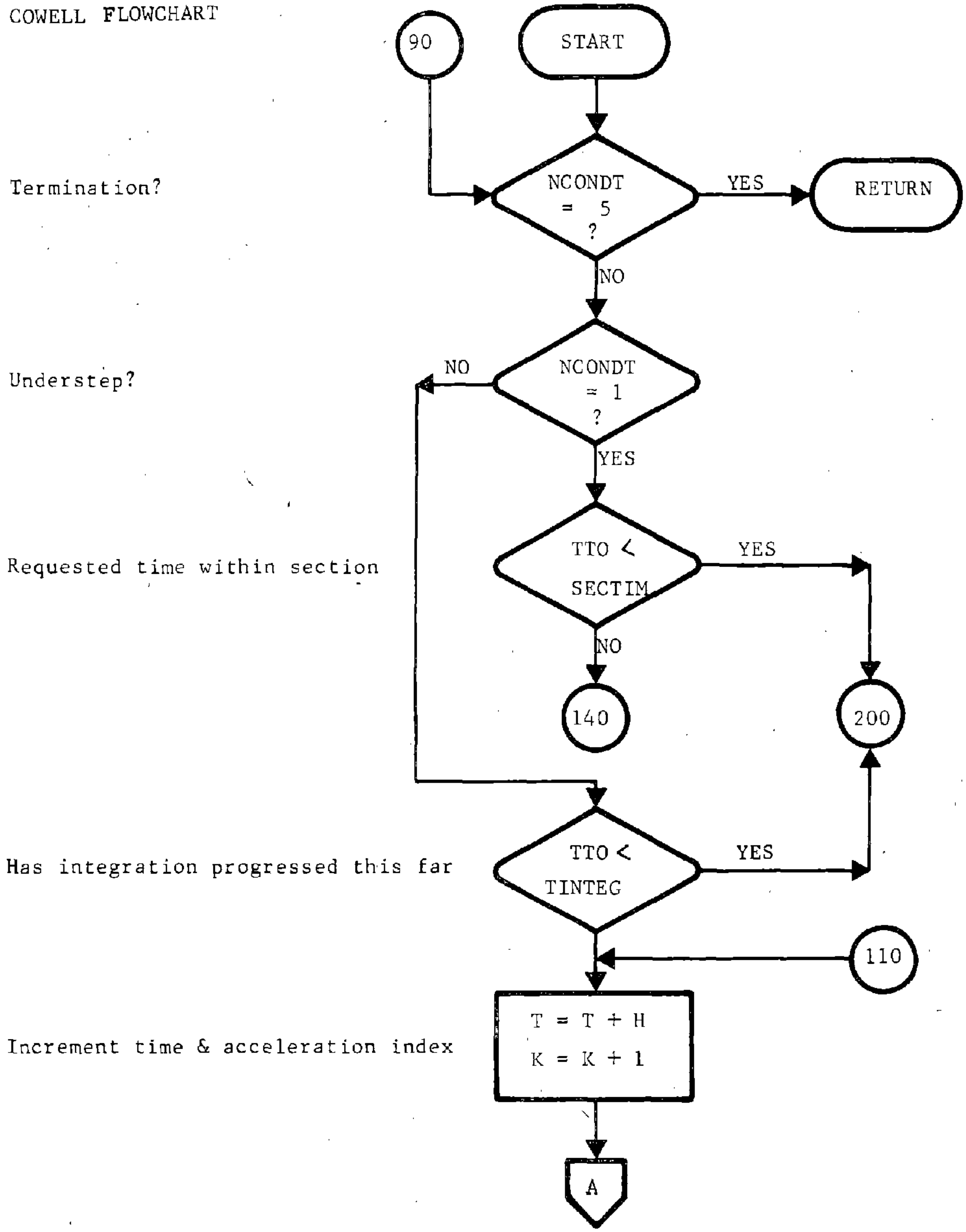
SUBROUTINES REQUIRED:

CSTEP
TESTH
INTP

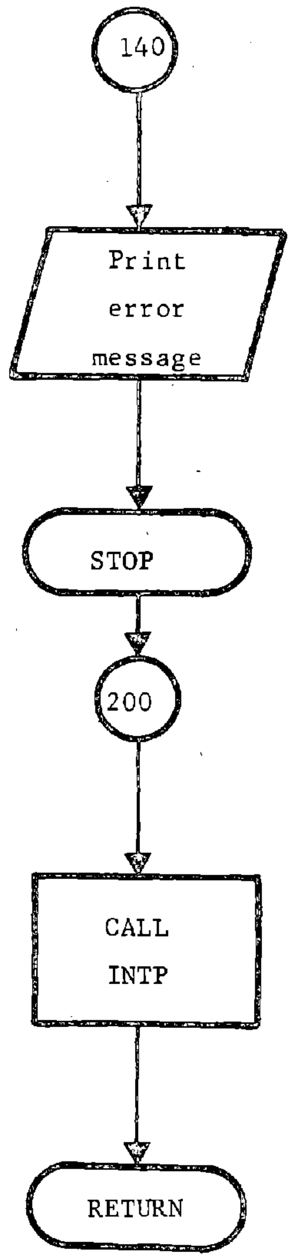
COMMON USED:

T	XVDD	SV1
K	IDON	SV2
H	IELEVN	
XDD	SX1	
NEQ	SX2	

COWELL FLOWCHART



Interpolate acceleration arrays for state and state partials at requested time



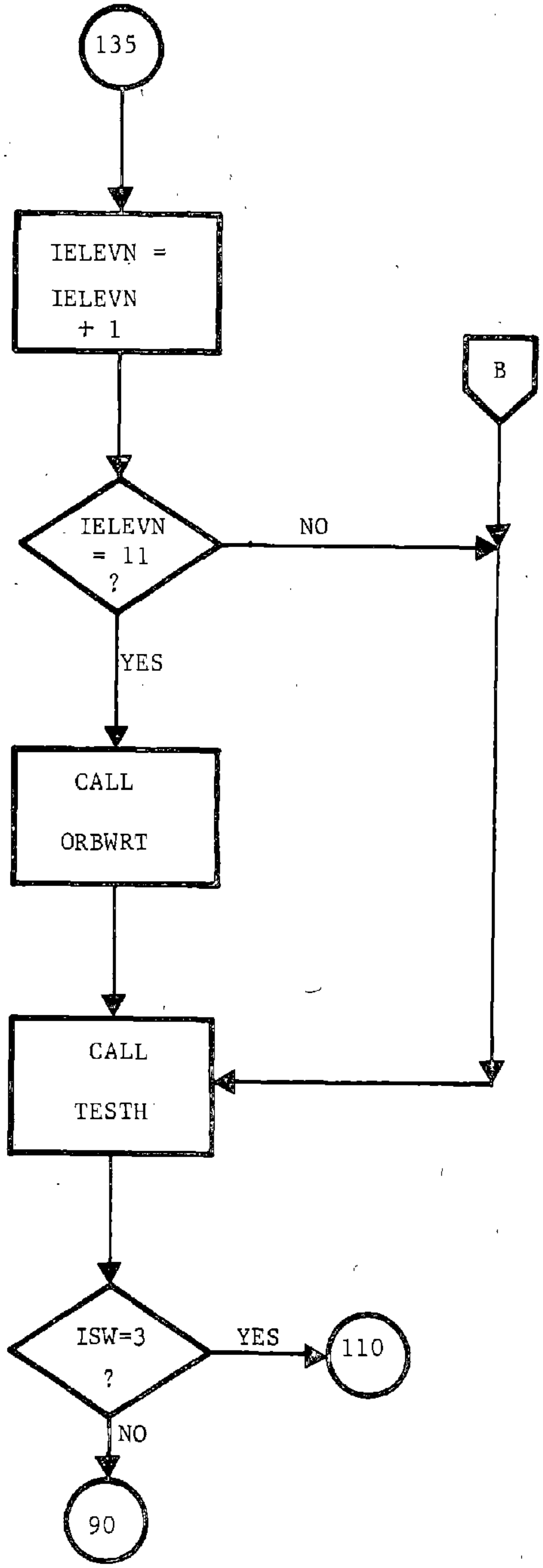
Increment point counter

Have eleven points accumulated?

Write last 11 acceleration points to orbit file

Check if step size should change

Has stepsize decreased?

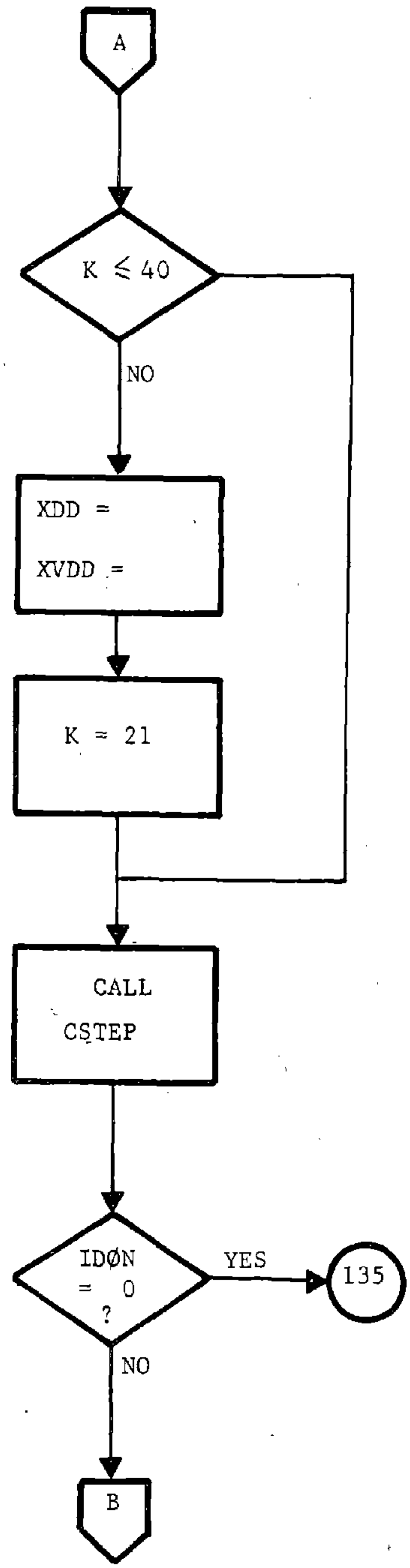


Acceleration arrays still have space?

Shift 20 old points out of acceleration arrays

Reset acceleration index

Skip writing an orbit file?



```
SUBROUTINE CSTART (ORBINT ENTRY POINT)

PURPOSE:  TO READ THE HEADER RECORD ON THE SEQUENTIAL ORBIT FILE

CALLING SEQUENCE:  CALL CSTART(NSEC,IERR)

ARGUMENTS:
          NSEC     I DESIRED TRAJECTORY SECTION NUMBER
          IERR     O ERROR FLAG
                     =1 NORMAL RETURN
                     =2 EOF DETECTED
                     =3 REQUESTED SECTION OUT OF RANGE
                     =4 REQUESTED TIME OUT OF RANGE

LOCAL SYMBOLS:
          IFRN       LOGICAL FILE NUMBER

COMMON COMPUTED:
          YMDIC      NSTATE     XDD         NSECTN
          HMSIC      KSTATE     SX1         NEQ
          AEINT      IPART      SX2
          SPINT      GM         XVDD
          PVINT      T          SV1
          OBLINT     H          SV2
```

```
FUNCTION DABSV

PURPOSE:  TO CALCULATE THE MAGNITUDE OF A VECTOR

CALLING SEQUENCE: RES=DABSV(A,N)

ARGUMENTS:
          A        I INPUT VECTOR
          N        I LENGTH OF VECTOR A
          DABSV    O MAGNITUDE OF VECTOR A

SUBROUTINES REQUIRED:
          NONE
```

SUBROUTINE DATA

PURPOSE: TO SET NECESSARY INITIAL VALUES AND DEFAULT VALUES FOR NAMELIST VARIABLES, TO READ INPUT DATA, TO TRANSLATE THESE INTO USEABLE INTERNAL VALUES, TO COMPUTE DIMENSIONS OF STATE TRANSITION AND COVARIANCE MATRIX PARTITIONS,TO ORDER MEASUREMENT AND EVENT SCHEDULES, AND TO PRINT THE INITIAL CONDITIONS.

CALLING SEQUENCE: CALL DATA

ARGUMENTS: NONE

SUBROUTINES SUPPORTED: STEAP-E %MAIN PROGRAM OF ERROR-ANALYSIS MODE<

SUBROUTINES REQUIRED: COVMAT CSTART GHA GDATA PECEQ SHIFT STVCPR SYMTRZ TINE ZERMAT

LOCAL SYMBOLS:

AP — ARRAY USED TO ORDER MEASUREMENTS AND EVENTS

AMIN — MINIMUM VALUE FOUND IN AP

D — TEMPORARY STORAGE FOR PRINTOUT
DD — TEMPORARY STORAGE FOR PRINTOUT

DUM — DUMMY VARIABLE
DUM1 — DUMMY VARIABLE

ECEQ — ROTATION MATRIX, ECLIPTIC TO EQUATORIAL

FNDT — JULIAN DATE OF FINAL TIME

ICNT — COUNTER USED IN SCHEDULING

IDAY — DAY OF THE MONTH OF FINAL TIME

IERPR — FLAG TO PRINT NAMELIST %IF SET<

IERR — FLAG FOR FILE-READER INITIALIZATION SUCCESS

IFCNRI — FLAG SET IF CONTROL COVARIANCE INPUT

IFGCOV — FLAG SET IF CONTROL COVARIANCES INPUT IN FORM OF STANDARD DEVIATIONS AND CORRELATIONS

IFPCOV — FLAG SET IF KNOWLEDGE COVARIANCES INPUT IN FORM OF STANDARD DEVIATIONS AND CORRELATIONS

IFVMRI	FLAG SET IF VARIATION MATRIX %RADA< INPUT
IHR	HOUR OF DAY OF FINAL TIME
IMIN	MINUTE OF HOUR OF FINAL TIME
IMO	MONTH OF YEAR OF FINAL TIME
IYR	YEAR OF FINAL TIME
IROW	INDEX USED IN SCHEDULING
LDAY	DAY OF MONTH OF INITIAL TIME ON FILE
LHR	HOUR OF DAY OF INITIAL TIME ON FILE
LMIN	MINUTE OF HOUR OF INITIAL TIME ON FILE
LMO	MONTH OF YEAR OF INITIAL TIME ON FILE
LYR	YEAR OF INITIAL TIME ON FILE
MEAS	TABLE OF MEASUREMENT TYPES REQUESTED
NDIMS	NUMBER OF COLUMNS OF AUGMENTED COVARIANCE
NENT	NUMBER OF CARDS IN MEASUREMENT SCHEDULE INPUT
SCHED	ARRAY OF MEASUREMENT SCHEDULE TIMES
SECI	SECONDS OF MINUTE OF FINAL TIME
SECL	SECONDS OF MINUTE OF INITIAL TIME ON FILE

```
COMMON COMPUTED/USED:  DATEJ    FNTM     IEVNT    MCODE    NDIM1
                       NDIM2    NDIM3    NDIM4    NEV      NMN
                       PVINT    TEV      TMN      TRTMB    TRTM1
                       XIG      XSL      XU       XV

COMMON COMPUTED:  CPLU     CXSU     CXSUG    CXSV     CXSVG    CXU
                  CXUG     CXV      CXVG     CXXS     CXXSG    EPS
                  EQEC     IAUG     IAUGW    IAUGDC   IAUGMC   P
                  PG       PHI      PHIG     PPLU     PS       PSG
                  PSPLU    TG       U0       V0       XG       XI

COMMON USED:  DELTM    IAUGIN   IGEN     NEV1     NEV2     NEV3     NEV4
              UNIVT    SAL      SLAT     SLON
```

```
FUNCTION DANGMD

PURPOSE:  TO MODULARIZE AN ANGLE ON TWO PI

CALLING SEQUENCE: RES=DANGMD(ANG)

ARGUMENTS:
          ANG      I THE ANGLE (RAD) TO BE MODULARIZED
          DANGMD   O (ANG) MOD TWO PI

SUBROUTINES REQUIRED:
          NONE
```

```
FUNCTION DANGV2

PURPOSE:  CALCULATE A DIRECTED ANGLE BETWEEN TWO VECTORS IN 3 SPACE

CALLING SEQUENCE: RES=DANGV2(A,B,REF)

ARGUMENTS:
          A        I FIRST VECTOR
          B        I SECOND VECTOR
          REF      I REFERENCE AXIS VECTOR
          DANGV2   O THE ANGLE FORMED BY ROTATING VECTOR A INTO
                     VECTOR B CLOCKWISE, LOOKING IN THE DIRECTION
                     OF THE VECTOR REF

SUBROUTINES REQUIRED:
          DUXV
          DABSV
          DDOT
```

```
SUBROUTINE DAVECT

PURPOSE:  VECTOR ADDITION

CALLING SEQUENCE: CALL DAVECT(A,B,N,C)

ARGUMENTS:
          A        I FIRST VECTOR
          B        I SECOND VECTOR
          N        I LENGTH OF VECTORS A AND B
          C        O VEC(A) + VEC(B)

SUBROUTINES REQUIRED:
          NONE
```

FUNCTION DDOT

PURPOSE: VECTOR DOT PRODUCT

CALLING SEQUENCE: RES=DDOT(A,B,N)

ARGUMENTS:

A	I	FIRST VECTOR
B	I	SECOND VECTOR
N	I	LENGTH OF VECTORS A AND B
DDOT	O	VEC(A) DOT VEC(B)

SUBROUTINES REQUIRED:
NONE

FUNCTION DDOTB

PURPOSE: TO RETURN THE DOT (INNER) PRODUCT OF TWO 3-VECTORS

CALLING SEQUENCE: ADOTB = DDOTB(A,B)

ARGUMENTS: A FIRST VECTOR
B SECOND VECTOR

SUBROUTINES REQUIRED: NONE

SUBROUTINE DMADD

PURPOSE: MATRIX ADDITION

CALLING SEQUENCE: CALL DMADD(A,B,C,I,J)

ARGUMENTS:

A	I	FIRST MATRIX
B	I	SECOND MATRIX
C	O	RESULT MATRIX = (A) + (B)
I	I	NUMBER OF ROWS IN MATRIX
J	I	NUMBER OF COLUMNS IN MATRIX

SUBROUTINES REQUIRED:
NONE

SUBROUTINE DMADD

PURPOSE: MATRIX ADDITION

CALLING SEQUENCE: CALL DMADD(A,B,C,I,J)

ARGUMENTS:

A	I	FIRST MATRIX
B	I	SECOND MATRIX
C	O	RESULT MATRIX = (A) + (B)
I	I	NUMBER OF ROWS IN MATRIX
J	I	NUMBER OF COLUMNS IN MATRIX

SUBROUTINES REQUIRED:
NONE

SUBROUTINE DMATPY

PURPOSE: MATRIX MULTIPLICATION

CALLING SEQUENCE: CALL DMATPY(A,B,C,N,M,IP)

ARGUMENTS:

A	I	FIRST MATRIX
B	I	SECOND MATRIX
C	O	RESULT MATRIX = (A) * (B)
N	I	NUMBER OF ROWS IN A
M	I	NUMBER OF COLUMNS IN A, NUMBER OF ROWS IN B
IP	I	NUMBER OF COLUMNS IN B

SUBROUTINES REQUIRED:
NONE

SUBROUTINE DMSUB (DMADD ENTRY POINT)

PURPOSE: MATRIX SUBTRACTION

CALLING SEQUENCE: CALL DMSUB(A,B,C,I,J)

ARGUMENTS:

A	I	FIRST MATRIX
B	I	SECOND MATRIX
C	O	RESULT MATRIX = (A) - (B)
I	I	NUMBER OF ROWS IN MATRIX
J	I	NUMBER OF COLUMNS IN MATRIX

SUBROUTINES REQUIRED:
NONE

```
SUBROUTINE DRD

PURPOSE:  TO COMPUTE LATITUDE AND LONGITUDE OF A VECTOR

CALLING SEQUENCE: CALL DRD (S,RA,DECL)

ARGUMENTS:
          S         I INPUT VECTOR
          RA        O RIGHT ASCENSION OR LONGITUDE OF S VECTOR
          DECL      O DECINATION OR LATITUDE OF S VECTOR

SUBROUTINES REQUIRED:
          NONE
```

```
SUBROUTINE DSHIFT

PURPOSE:  TO SHIFT ONE VECTOR INTO ANOTHER

CALLING SEQUENCE: CALL DSHIFT (A,N,B)

ARGUMENTS:
          A         I ADDRESS OF FIRST DOUBLE WORD TO BE SHIFTED
          N         I NUMBER OF DOUBLE WORDS TO BE SHIFTED
          B         O ADDRESS OF FIRST DOUBLE WORD IN RECEIVING VECTOR

SUBROUTINES REQUIRED:
          NONE
```

```
SUBROUTINE DSVECT (DAVECT ENTRY POINT)

PURPOSE:  VECTOR SUBTRACTION

CALLING SEQUENCE: CALL DSVECT(A,B,N,C)

ARGUMENTS:
          A         I FIRST VECTOR
          B         I SECOND VECTOR
          N         I LENGTH OF VECTORS A AND B
          C         O VEC(A) - VEC(B)

SUBROUTINES REQUIRED:
          NONE
```

SUBROUTINE DUNIT

PURPOSE: TO UNITIZE A VECTOR

CALLING SEQUENCE: CALL DUNIT (A,N,B)
ARGUMENTS:

A	I	INPUT VECTOR
N	I	LENGTH OF VECTOR
B	O	UNIT VECTOR IN THE DIRECTION OF A

SUBROUTINES REQUIRED:
DABSV

SUBROUTINE DUXV

PURPOSE: TO FORM A VECTOR CROSS PRODUCT

CALLING SEQUENCE: CALL UXV(A,B,C)

ARGUMENTS:

A	I	FIRST VECTOR
B	I	SECOND VECTOR
C	O	RESULT VECTOR = (A) CROSS (B)

SUBROUTINES REQUIRED:
NONE

SUBROUTINE DVCOMB

PURPOSE: TO COMBINE (ADD) TWO VECTORS, EACH MULTIPLIED BY SCALARS

CALLING SEQUENCE: CALL DVCOMB(A,SA,B,SB,C)

ARGUMENTS:

A	I	FIRST VECTOR
SA	I	SCALAR WHICH IS APPLIED TO VECTOR A
B	I	SECOND VECTOR
SB	I	SCALAR WHICH IS APPLIED TO VECTOR B
C	O	RESULT VECTOR = SA*(A) + SB*(B)

SUBROUTINES REQUIRED:
NONE

```
SUBROUTINE DVECRD

PURPOSE:  TO COMPUTE A UNIT VECTOR FROM ITS RIGHT ASCENSION AND
          DECLINATION

CALLING SEQUENCE: CALL DVECRD(RA,DECL,S)

ARGUMENTS:
          RA        I RIGHT ASCENSION OR LONGITUDE (RAD)
          DECL      I DECLINATION OR LATITUDE (RAD)
          S         O OUTPUT VECOTR

SUBROUTINES REQUIRED:  NONE
```

```
FUNCTION DVMAG

PURPOSE:  CALCULATE THE MAGNITUDE OF A 3-VECTOR

CALLING SEQUENCE:  AMAG = DVMAG(A)

ARGUMENTS:  A  VECTOR

SUBROUTINES REQUIRED:  NONE
```

```
SUBROUTINE DVSDIV (SAVECT ENTRY POINT)

PURPOSE:  VECTOR SCALAR DIVISION

CALLING SEQUENCE: CALL DVSDIV(A,SA,N,C)

ARGUMENTS:
          A         I INPUT VECTOR
          SA        I SCALAR BY WHICH VECTOR A IS DIVIDED
          N         I LENGTH OF VECTOR A
          C         O VEC(A) / SA

SUBROUTINES REQUIRED:  NONE
```

```
SUBROUTINE DVSMLT (SAVECT ENTRY POINT)

PURPOSE:  VECTOR SCALAR MULTIPLICATION

CALLING SEQUENCE: CALL DVSMLT(A,SA,N,C)

ARGUMENTS:
          A         I INPUT VECTOR
          SA        I SCALAR BY WHICH VECTOR A IS MULTIPLIED
          N         I LENGTH OF VECTOR A
          C         O SA * VEC(A)

SUBROUTINES REQUIRED:  NONE
```

SUBROUTINE DVSTAT

PURPOSE: COMPUTE AND PRINT STATISTICAL PARAMETERS OF A TRIM MANEUVER

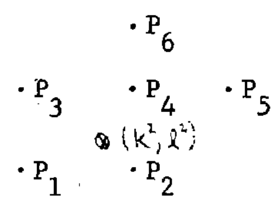
CALLING SEQUENCE: CALL DVSTAT(S,E,V,T,TR,RHO,SMX)

```
ARGUMENTS:  S     INPUT MATRIX, GAMMA*P*GAMMA(TRANSPOSE)
            E     ARRAY OF EIGENVALUES OF S
            V     ARRAY OF EIGENVECTORS CORRESPONDING TO E
            T     ANNIHILATION LIMIT (IF(E(I).LT.T)E(I)=ZERO)
            TR    TRACE OF S
            RHO   MEAN VALUE OF DELTA-V
            SMX   MAXIMUM EIGENVALUE IN E

LOCAL SYMBOLS:  D     TABLE OF PERCENTILE LEVELS AND DELTA-V MAGNITUDES
                DV    ARRAY OF CONSTANTS USED TO COMPUTE MAGNITUDES
                IK    *
                IK1   *** INDICES USED IN LAGRANGIAN 6-POINT INTER-
                IL    *        POLATION
                IL1   *
                K2    RATIO, SMD/SMX
                L2    RATIO, SMN/SMX
                SD    STANDARD DEVIATION OF DELTA-V VALUES
                SMD   MIDDLE  EIGENVALUE IN E
                SMN   MINIMUM EIGENVALUE IN E
                STR   SQUARE ROOT OF TR
```

DVSTAT Analysis

The Lee-Boain analytic solution for V statistics involves a hypergometric function, and is described in detail in Reference 6. For this subroutine, a table DV has been generated from their solution which is used to calculate the mean and standard deviation and values for the 90, 99, 99.9 and 99.99 percentile levels in a much less time. The subroutine JACØBI obtains the eigen values and eigenvectors of the input S matrix. The ratios of the middle and smallest eigenvalues to the largest eigenvalue, k^2 and l^2 respectively, are determined and used in a Lagrangian 6-point interpolation from table DV. The six points are located as follows



where $P_1 = \left(\frac{[10k^2]}{10}, \frac{[10\ell^2]}{10}\right)$, $[X]$ = greatest integer less than or equal to X

and the other points are at appropriate .1 intervals.

Define

$$p = [10k^2 + 1] - 10k^2$$

$$q = [10\ell^2 + 1] - 10\ell^2$$

Use Lagrangian 6 point interpolation:

$$\begin{aligned} f(k^2, \ell^2) = {} & pq \cdot f(p_1) + q(q-2p+1) \cdot f(p_3) \,/\, 2 \\ & + p(p-2q+1) \cdot f(p_3) \,/\, 2 \\ & + (1+pq-p^2-q^2) \cdot f(p_4) \\ & + p(p-1) \cdot f(p_5) \,/\, 2 \\ & + q(q-1) \cdot f(p_6) \,/2 \end{aligned}$$

This necessitates data points at .1 intervals over the range $0 \leq k^2 \leq 1.1$, $0 \leq \ell^2 \leq k^2$.

Points outside this range, even though they might figure in the interpolation theoretically, are unnecessary because either p or q or both become 1, and the corresponding terms drop out.

After values have been so generated for μ, $\Delta v_{.9}$, $\Delta v_{.99}$, $\Delta v_{.999}$, $\Delta v_{.9999}$, the values are multiplied by the square root of the trace, which is normalized to one in generating the data points. The standard deviation is calculated from the classic formula, $\sigma = (tr - \mu^2)^{\frac{1}{2}}$.

SUBROUTINE DYNO

PURPOSE: COMPUTE ASSUMED AND ACTUAL DYNAMIC NOISE COVARIANCE MATRIX IN THE ERROR ANALYSIS PROGRAM

CALLING SEQUENCE: CALL DYNO(ICODE)

ARGUMENT: ICODE I =0 ASSUMED
=1 ACTUAL

SUBROUTINES SUPPORTED: ERRANN SETEVN GUIDM PRED

LOCAL SYMBOLS: D2 SQUARE OF (DELTM*TM)

COMMON COMPUTED: Q QPR

COMMON USED: DELTM DNCN IDNF TM
IGDNF GDNCN

DYNØ Analysis

Subroutine DYNØ evaluates the assumed dynamic covariance matrix Q over the time interval $t = t_{k+1} - t_k$ if ICØDE = 0. If ICØDE = 1 the actual dynamic noise covariance matrix Q' is evaluated over the same interval. In either case the dynamic noise covariance matrix is assumed to have the form

$$Q = \text{diag}\ (\tfrac{1}{4} K_1\ \Delta t^4,\ \tfrac{1}{4} K_2\ \Delta t^4,\ \tfrac{1}{4} K_3\ \Delta t^4,\ K_1\ \Delta t^2,\ K_2\ \Delta t^2,\ K_3\ \Delta t^2)$$

where dynamic noise constants K_1, K_2, and K_3 have units of km^2/s^4. To compute the actual dynamic noise covariance matrix Q', we simply replace K_1, K_2, and K_3 with the actual dynamic noise constants K_1', K_2', and K_3', respectively.

SUBROUTINE DXB

PURPOSE: CALCULATE THE CROSS PRODUCT OF TWO 3-VECTORS

CALLING SEQUENCE: CALL DXB(AA,BB,VECPR)

ARGUMENTS: AA FIRST INPUT VECTOR
BB SECOND INPUT VECTOR
VECPR VECTOR PRODUCT

SUBROUTINES REQUIRED: NONE

SUBROUTINE DZERO

PURPOSE: TO GENERATE A ZERO VECTOR

CALLING SEQUENCE: CALL DZERO(A,N)

ARGUMENTS:
A I ADDRESS OF FIRST DOUBLE WORD TO BE ZEROED
N N NUMBER OF DOUBLE WORDS TO BE ZEROED

SUBROUTINES REQUIRED:
NONE

```
SUBROUTINE ECOMP

PURPOSE:  TO COMPUTE DIFFERENTIAL TRANSFORMATION RELATING TARGET
          VARIABLES TO STATE

CALLING SEQUENCE:  CALL ECOMP(XX,AA,BB)

ARGUMENTS
          XX        I CURRENT STATE VECTOR
          AA        O PARTIALS OF TARGETS WITH RESPECT TO POSITION
          BB        O PARTIALS OF TARGETS WITH RESPECT TO VELOCITY

LOCAL SYMBOLS:
          XPERT      TEMPORARY PERTURBATION VALUE
          TFP        TIME FROM PERIGEE
          E1         SEMI-MAJOR AXIS
          E2         ECCENTRICITY
          ATN        3 VECTOR OF TARGETS CORRESPONDING TO NEGATIVE
                     PERTURBATION MADE TO XX
          ATP        3 VECTOR OF TARGETS CORRESPONDING TO POSITIVE
                     PERTURBATION MADE TO XX

SUBROUTINES REQUIRED:
          CAREL

COMMON USED:
          PERT
          TM
          SMU(4)
```

SUBROUTINE EIGHY

PURPOSE: TO CONTROL THE COMPUTATION OF EIGENVALUES, EIGENVECTORS, AND HYPERELLIPSOIDS.

CALLING SEQUENCE: CALL EIGHY(VEIG,FOX,HARG,IFMT)

ARGUMENT: VEIG I MATRIX TO BE DIAGONALIZED

FOX I FINAL OFF-DIAGONAL ANNIHILATION VALUE

HARG I MATRIX FOR WHICH THE HYPERELLIPSOID IS TO BE COMPUTED

IFMT I FORMAT FLAG
=1, PRINT POSITION EIGENVALUE TITLE
=2, PRINT VELOCITY EIGENVALUE TITLE
=3, PRINT EIGENVALUE TITLE

SUBROUTINES SUPPORTED: ATCEGV MOMENT PRED GENGID GUIDM

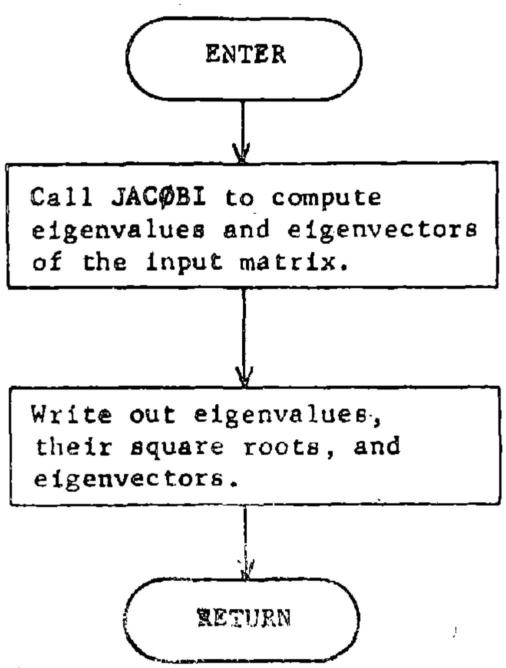
SUBROUTINES REQUIRED: JACOBI

LOCAL SYMBOLS: EGVCT EIGENVECTOR MATRIX

EGVL EIGENVALUE MATRIX

OUT SQUARE ROOTS OF EIGENVALUES

EIGHY Flow Chart



SUBROUTINE EPHGT

PURPOSE: TO RETRIEVE FROM THE DIRECT ACCESS SLP FILE, THE STATE VECTOR OF A PLANET WITH RESPECT TO THE SUN AT AN ARBITRARY JULIAN DATE

CALLING SEQUENCE: CALL EPHGT(IP,DJ,R,V)

ARGUMENTS:

IP	I	PLANET NUMBER (1=SUN, 2=MERCURY, ETC)
DJ	I	JULIAN DATE
R	O	RADIUS VECTOR FROM SUN TO PLANET IP
V	O	VELOCITY VECTOR OF PLANET IP WITH RESPECT TO THE SUN

LOCAL SYMBOLS:

IPGSFC	VECTOR CORRELATING NOMNAL PLANET NUMBERING SYSTEM WITH GTDS PLANET NUMBERING CONVENTION
ARRAY	TEMPORARY TRANSMISSION ARRAY BETWEEN EPHGT AND SUBROUTINE EVAL
IARRAY	TEMPORARY TRANSMISSION ARRAY BETWEEN EPHGT AND SUBROUTINE EVAL

COMMON USED:

YMDIC
HMSIC
IND(14)

EPHGT Analysis

Subroutine EPHGT is used by ERRAN and NOMNAL to retrieve heliocentric-ecliptic planetary state vectors from the solar/lunar/planetary direct access ephemeris file. This is accomplished by saving and resetting the year, month, day and hours, minute, seconds variables (for initial conditions) and a dummy gravitating bodies vector with the sun as the central body and the planet of interest as the only non-central body is then generated. Subroutine EVAL is then called to generate these vectors.

PROGRAM ERRAN

PURPOSE: TO CONTROL THE COMPUTATIONAL FLOW THROUGH THE BASIC CYCLE (MEASUREMENT PROCESSING) AND ALL EVENTS IN THE ERROR ANALYSIS MODE.

SUBROUTINES SUPPORTED: ERRON

SUBROUTINES REQUIRED:	SCHED	NTM	PSIM	DYNO	TRAKM
	MENO	GNAVM	PRINT3	SETEVN	GUIDM
	MEAN	GPRINT	PRED	GENGID	

LOCAL SYMBOLS:

ICODE	EVENT CODE
IPRN	MEASUREMENT COUNTER FOR PRINTING
NEVENT	EVENT COUNTER
TRTM2	TIME OF THE MEASUREMENT

COMMON COMPUTED/USED:	ICODE	MCNTR	RI	TEVN	TRTM1
	XF	XI			
COMMON COMPUTED:	DELTM				
COMMON USED:	FNTM	IEVNT	IPRINT	ISTMC	NEV
	NMN	NR	NTMC	RF	TEV
	KPRINT				

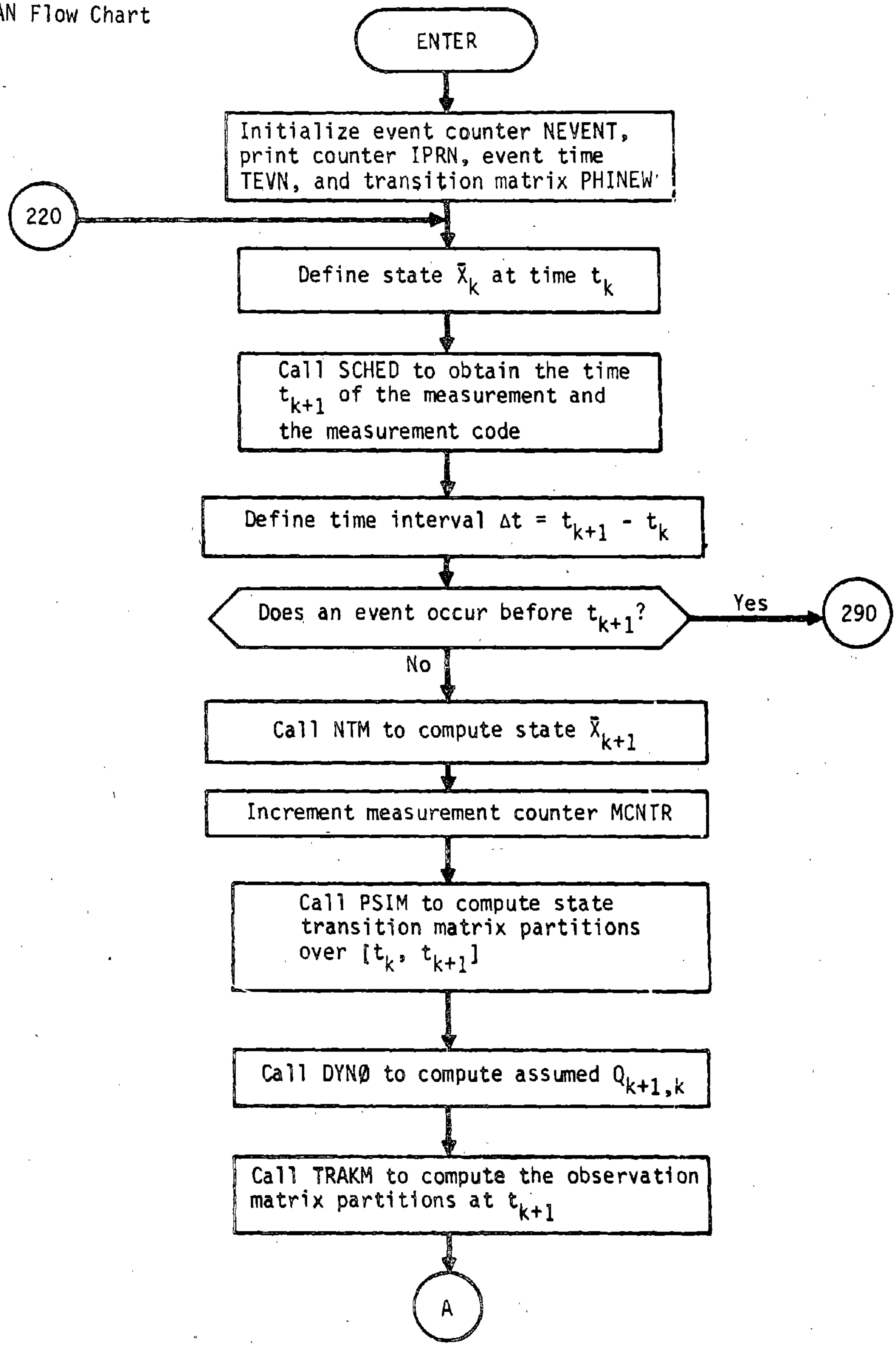
ERRAN Analysis

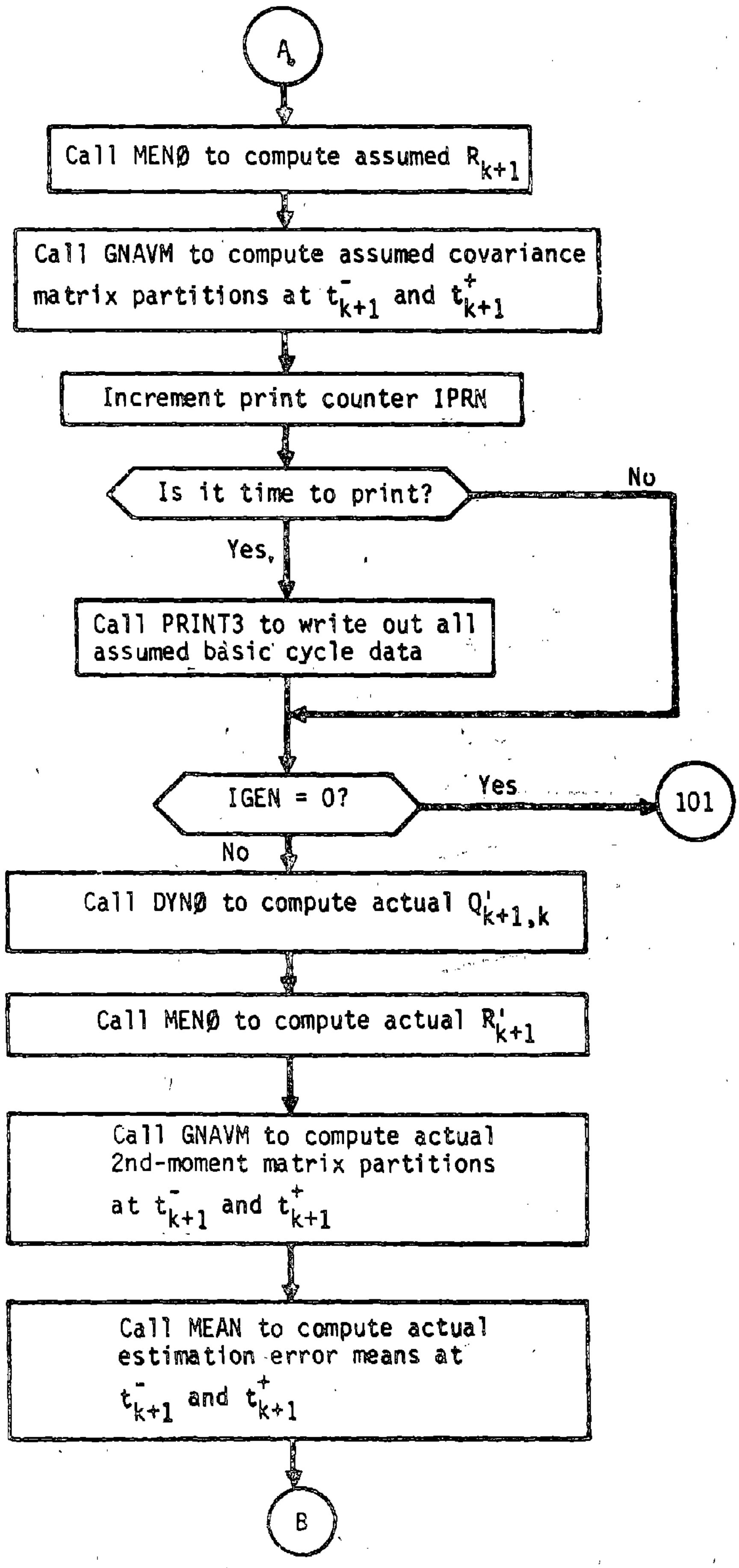
Subroutine ERRAN controls the computational flow through the basic cycle (measurement processing) and all events in the error analysis/generalized covariance analysis program.

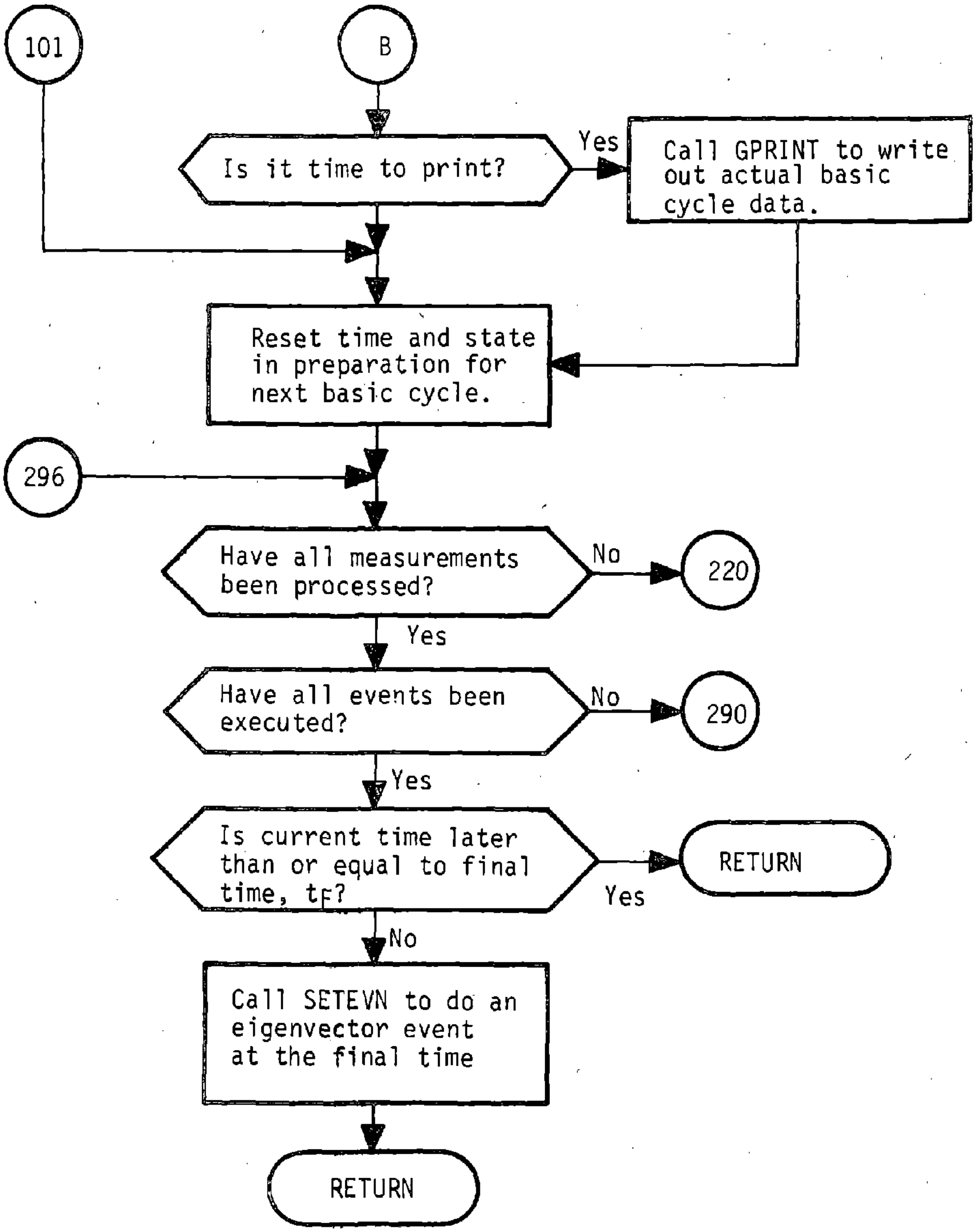
In the basic cycle the first task of ERRAN is to control the generation of the targeted nominal spacecraft state $\bar{X}_{k+1}$ at time t_{k+1}, given the state $\bar{X}_k$ at time t_k. Then calling PSIM, DYNØ, TRAKM, and MENØ, successively, ERRAN controls the computation of all matrix information required by subroutine GNAVM to compute the actual and assumed knowledge covariance matrix partitions at time t^+_{k+1} immediately following the measurement.

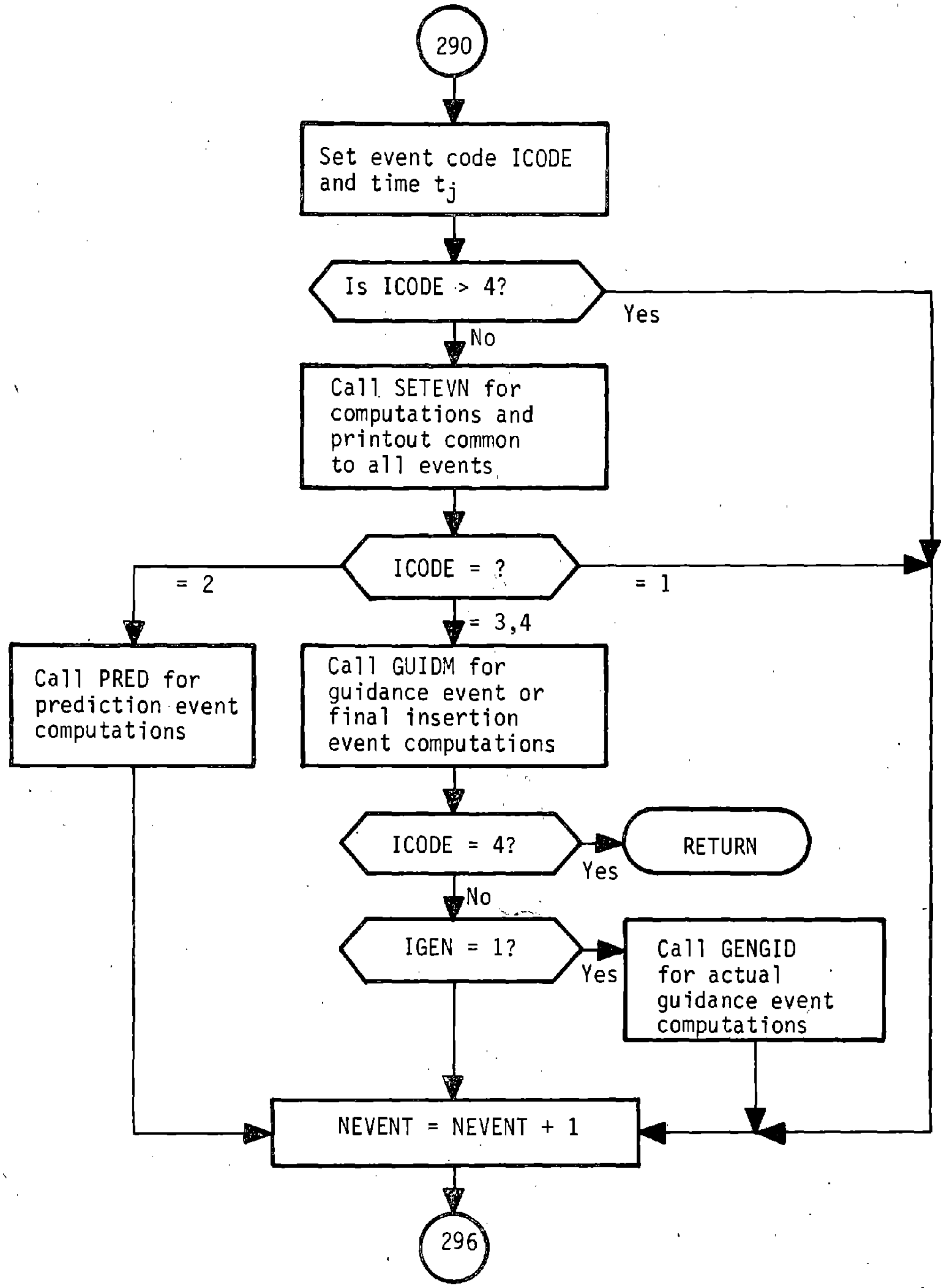
At an event, ERRAN simply calls the proper event subroutine or overlay where all required computations are performed.

ERRAN Flow Chart




A
Call MENØ to compute assumed R_{k+1}
Call GNAVM to compute assumed covariance matrix partitions at t_{k+1}^- and t_{k+1}^+
Increment print counter IPRN
Is it time to print?
No
Yes
Call PRINT3 to write out all assumed basic cycle data
IGEN = 0?
Yes
101
No
Call DYNØ to compute actual $Q'_{k+1,k}$
Call MENØ to compute actual R'_{k+1}
Call GNAVM to compute actual 2nd-moment matrix partitions at t_{k+1}^- and t_{k+1}^+
Call MEAN to compute actual estimation error means at t_{k+1}^- and t_{k+1}^+
B


101
B
Is it time to print?
Yes
Call GPRINT to write out actual basic cycle data.
Reset time and state in preparation for next basic cycle.
296
Have all measurements been processed?
No
220
Yes
Have all events been executed?
No
290
Yes
Is current time later than or equal to final time, t_F?
Yes
RETURN
No
Call SETEVN to do an eigenvector event at the final time
RETURN


290
Set event code ICODE and time t_j
Is ICODE > 4?
Yes
No
Call SETEVN for computations and printout common to all events
ICODE = ?
= 2
= 1
= 3,4
Call PRED for prediction event computations
Call GUIDM for guidance event or final insertion event computations
ICODE = 4?
Yes
RETURN
No
IGEN = 1?
Yes
Call GENGID for actual guidance event computations
NEVENT = NEVENT + 1
296

SUBROUTINE EULMX

PURPOSE: TO COMPUTE THE MATRIX REQUIRED TO DEFINE TRANSFORMATIONS FROM ONE COORDINATE SYSTEM TO ANOTHER.

CALLING SEQUENCE: CALL EULMX(ALP,NN,BET,MM,GAM,LL,P)

ARGUMENT:

ALP	I	FIRST ROTATION ANGLE (RADIANS)
NN	I	FIRST AXIS OF ROTATION
BET	I	SECOND ROTATION ANGLE (RADIANS)
MM	I	SECOND AXIS OF ROTATION
GAM	I	THIRD ROTATION ANGLE (RADIANS)
LL	I	THIRD AXIS OF ROTATION
P(3,3)	O	TRANSFORMATION MATRIX

SUBROUTINES REQUIRED: NONE

LOCAL SYMBOLS:

A	INTERMEDIATE ROTATION MATRIX
ALPHA	TEMPORARY LOCATION FOR EACH OF THE ROTATION ANGLES: ALP, BET, AND GAM
D	INTERMEDIATE PRODUCT MATRIX
F	TRANSFORMATION MATRIX FOR ANGLE ALP
G	TRANSFORMATION MATRIX FOR ANGLE BET
H	TRANSFORMATION MATRIX FOR ANGLE GAM
N	COUNTER SHOWING NUMBER OF COORDINATE AXES FOR WHICH CALCULATIONS REMAIN
NAXIS	TEMPORARY LOCATION FOR EACH OF THE AXES OF ROTATION: NN,MM, AND LL

SUBROUTINE GAIN1

PURPOSE: TO COMPUTE THE KALMAN GAIN MATRICES

CALLING SEQUENCE: CALL GAIN1(NR,AJ,AKW,SW,IEND)

ARGUMENTS: NR I NUMBER OF ROWS IN THE OBSERVATION MATRIX

AJ I MEASUREMENT RESIDUAL COVARIANCE AND ITS INVERSE

AKW I INTERMEDIATE ARRAY

SW I INTERMEDIATE ARRAY

IEND I NR-1

SUBROUTINES SUPPORTED: GNAVM

SUBROUTINES REQUIRED: MATIN

LOCAL SYMBOLS: DUM INTERMEDIATE VECTOR

XJ INTERMEDIATE ARRAY

SUM INTERMEDIATE VARIABLE

COMMON COMPUTED: AK S

COMMON USED: ONE HALF ZERO

GAIN1 Analysis

Subroutine GAIN1 computes the Kalman-Schmidt filter gain matrices K_{k+1} and S_{k+1} that are used in subroutines GNAVM and NAVM to update estimation error covariance matrices after a measurement has been processed.

The measurement residual covariance matrix J_{k+1} and the auxiliary matrices A_{k+1} and B_{k+1} are assumed to be available (from GNAVM or NAVM) when GAIN1 is called. Subroutine GAIN1 then evaluates the following equations to determine the filter gain matrices:

$$K_{k+1} = A_{k+1} J_{k+1}^{-1} \tag{1}$$

$$S_{k+1} = B_{k+1} J_{k+1}^{-1} \tag{2}$$

SUBROUTINE GDATA

PURPOSE: TO INITIALIZE GENERALIZED COVARIANCE QUANTITIES

CALLING SEQUENCE: CALL GDATA

SUBROUTINES SUPPORTED: DATA

COMMON COMPUTED/USED:

EU	EV	EVA	EVB	EVK
EVS	EW	EXI	EXSI	EXST
EXT	GCUV	GCUW	GCVW	GCXSU
GCXSUG	GCXSV	GCXSVG	GCXSW	GCXSWG
GCXU	GCXUG	GCXV	GCXVG	GCXW
GCXWG	GCXXS	GCXXSG	GDNCN	GMNCN
GP	GPG	GPS	GPSG	GU
GV	GW	IDNF	VARA	VARB
VARK	VARS			

COMMON USED:

CXSU	CXSV	CXU	CXV	CXXS
DNCN	IDNF	MNCN	NDIM1	NDIM2
NDIM3	NDIM4	P	PS	SIGALP
SIGBET	SIGPRO	SIGRES	TG	U0
V0	ZERO			

SUBROUTINE GENGID

PURPOSE: TO GENERATE THE ENSEMBLE STATISTICS OF THE ACTUAL COMMANDED VELOCITY CORRECTION, THE ACTUAL EXECUTION ERROR AND THE ACTUAL TARGET MISS

CALLING SEQUENCE: CALL GENGID

SUBROUTINES SUPPORTED: ERRAN

SUBROUTINES REQUIRED: SAVMAT DYNO GNAVM MEAN MOMENT EIGHY GQCOMP ATCEGV JACOBI DVSTAT

LOCAL SYMBOLS:

AMAX	INTERMEDIATE VARIABLE
ATC	ACTUAL TARGET CONDITION 2ND MOMENT MATRIX
B	INTERMEDIATE VARIABLE
BBBB	BLANK LABEL ARRAY
C	INTERMEDIATE VARIABLE
DELTM	TIME DIFFERENCE
EBDVB	MAGNITUDE OF ACTUAL STATISTICAL DELTA-V
EDVN	MEAN OF ACTUAL COMMANDED VELOCITY CORRECTION
EGM	MAGNITUDE OF EIGENVECTOR CORRESPONDING TO MAXIMUM EIGENVALUE
EGVCT	EIGENVECTOR ARRAY
EGVL	EIGENVALUE VECTOR
ELAB	LABEL
EXIS	STORAGE FOR EXI
EXSIS	STORAGE FOR EXSI
EXTS	STORAGE FOR EXT
EXV	ACTUAL STATISTICAL DELTA-V
GAP	ACTUAL VELOCITY CORRECTION 2ND MOMENT MATRIX
GPSAVE	STORAGE FOR GP

GTG	TIME OF ACTUAL GUIDANCE EVENT
IFLAG	=1 BEFORE GUIDANCE EVENT =2 AFTER GUIDANCE EVENT
III	INDEX DEPENDING ON GUIDANCE EVENT TYPE
MAP	INDEX OF MAXIMUM EIGENVALUE
PEIG	INTERMEDIATE ARRAY
Q	ACTUAL EXECUTION ERROR 2ND MOMENT MATRIX
ROW	INTERMEDIATE VECTOR
S	INTERMEDIATE ARRAY
SUM	INTERMEDIATE VARIABLE
U	ACTUAL COMMANDED VELOCITY CORRECTION
VEIG	INTERMEDIATE VECTOR
ZLAB	LABEL
ZV	ACTUAL EXECUTION ERROR MEANS
ZZ	INTERMEDIATE VARIABLE

COMMON COMPUTED/USED:

DUMMYQ	EXI	EXMEAN	EXSI	EXT
GCXSUG	GCXSVG	GCXSWG	GCXUG	GCXVG
GCXWG	GCXXSG	GP	GPG	GPSG
XLAB				

COMMON USED:

ADA	DVUP	EE	EEE	EU
EV	EW	EXST	FOP	FOV
GA	GCXSU	GCXSV	GCXSW	GCXU
GCXV	GCXW	GCXXS	GPS	GU
GV	GW	IGP	IGUID	II
NDIM1	NDIM2	NDIM3	NDIM4	PI
QPR	RPR	TEVN	TG	TINJ
XIG	XSL	XU	XV	

GENGID Analysis

Subroutine GENGID controls the execution of generalized guidance events. Generalized guidance has been extended to all guidance options defined for subroutine GUIDM except for final insertion.

Unlike GUIDM, which computes target dispersions and fuel budgets based on filter-generated statistics, subroutine GENGID computes target dispersions and fuel budgets based on actual statistics. In other words, the generalized covariance technique as applied to the guidance process is programmed in GENGID. The required equations are summarized below.

Before the guidance event at time t_j can be executed, it is necessary to propagate the actual control mean and control 2nd-moment matrix partitions forward to t_j from the previous guidance event at time t_{j-1}. The control mean propagates according to

$$\bar{x}_j^{\prime -} = \Phi\, \bar{x}_{j-1}^{\prime +} + \theta_{xx_s}\, \bar{x}_{s_o}^{\prime} + \theta_{xu}\, \bar{u}_o^{\prime} + \theta_{xw}\, \bar{w}_o^{\prime} \qquad (1)$$

where Φ, θ_{xx_s}, θ_{xu}, and θ_{xw} are state transition matrix partitions over the interval $\left[t_{j-1}, t_j\right]$, and $\bar{x}^{\prime}$, $\bar{x}_s^{\prime}$, $\bar{u}^{\prime}$, and $\bar{w}^{\prime}$ denote actual position/velocity and solve-for, dynamic-consider, and ignore parameter deviation means. The notation $(\ )^{\prime}$ indicates actual values as opposed to the unprimed assumed values, while $(\ )^{-}$ and $(\ )^{+}$ indicate values immediately before and after the execution of the guidance event, respectively. The actual control position/velocity 2nd-moment matrix is defined by

$$P_{c_j} = E\left[x_j^{\prime}\, x_j^{\prime T}\right]. \qquad (2)$$

The remaining control 2nd-moment matrix partitions are defined similarly. The propagation equations appearing in subroutine GNAVM are used to propagate the control 2nd-moment matrix partitions over the interval $\left[t_{j-1}, t_j\right]$.

The actual target state deviation $\delta\tau_j'$ is related to the actual state deviation x_j' at time t_j according to

$$\delta\tau_j' = \eta_j \, x_j' \tag{3}$$

where η_j is the variation matrix for the appropriate midcourse guidance policy. The mean of $\delta\tau_j'$ is given by

$$E\left[\delta\tau_j'\right] = \eta_j \, E\left[x_j'\right]. \tag{4}$$

The statistical target dispersions are represented by the actual target condition 2nd-moment matrix W_j', which is defined as

$$W_j' = E\left[\delta\tau_j' \, \delta\tau_j'^T\right]. \tag{5}$$

Substitution of equation (3) into equation (5) yields

$$W_j' = \eta_j \, P_{c_j}' \, \eta_j^T . \tag{6}$$

Equations (4) and (6) are evaluated immediately before and after the guidance correction to determine how much the target errors have actually been reduced by the velocity correction at t_j.

The actual commanded velocity correction 2nd-moment matrix is defined by

$$S_j' = E\left|\Delta\hat{V}_j' \, \Delta\hat{V}_j'^T\right| \tag{7}$$

where the actual commanded velocity correction is given by

$$\Delta\hat{V}_j' = \Gamma_j \, \hat{x}_j' = \Gamma_j \left(x_j' + \tilde{x}_j'\right). \tag{8}$$

The guidance matrix Γ_j corresponds to the appropriate linear midcourse guidance policy. The equation used to evaluate S_j' is given by

$$S_j' = \Gamma_j \left(P_{c_j}' - SP_{k_j}'\right) \Gamma_j^T \tag{9}$$

where s is a scalar input by the analyst, generally $0. \leq s \leq 1.$, and

where all $E\left[x_j' \tilde{x}_j'^T\right]$ terms have been neglected in the derivation of equation (9).

The mean of the actual commanded velocity correction is obtained by applying the expectation operator to equation (8):

$$E\left[\Delta V_j'\right] = \Gamma_j \left\{E\left[x_j'\right] + E\left[\tilde{x}_j'\right]\right\}. \tag{10}$$

Since this equation gives no useful information for fuel-sizing studies, the Hoffman-Young formula will be used to evaluate $E\left[\left|\Delta V_j'\right|\right]$

$$E\left|\left|\Delta \hat{V}_j'\right|\right| = \sqrt{\frac{2A}{\pi}}\left(1 + \frac{B\ (\pi - 2)}{A^2\sqrt{5.4}}\right) \tag{11}$$

where

$$A = \text{trace } S_j'$$

$$B = \lambda_1' \lambda_2' + \lambda_1' \lambda_3' + \lambda_2' \lambda_3',$$

and λ_1', λ_2', and λ_3' are the eigenvalues of the 2nd-moment matrix S_j'. If this mean is vanishingly small, the Lee-Boain analysis is used to obtain the statistical parameters, including the effective ΔV. Otherwise the actual effective or statistical ΔV is defined as

$$\text{"}E\left[\Delta \hat{V}_j'\right]\text{"} = E\left[\left|\Delta \hat{V}_j'\right|\right] \cdot \alpha_j' \tag{12}$$

where α_j' denotes a unit vector in the most likely direction of the velocity correction. The most likely direction is assumed to be aligned with the eigenvector associated with the maximum eigenvalue of S_j'.

With "$E\left[\Delta \hat{V}_j'\right]$" available, the actual execution error statistics can be computed (by calling subroutine GQCØMP). These are the actual execution error mean $E\left[\delta \Delta V_j'\right]$ and 2nd-moment maxtrix $\tilde{Q}_j'$ defined as

$$\tilde{Q}_j' = E\left[\delta \Delta V_j' \ \delta \Delta V_j'^T\right]. \tag{13}$$

It remains to summarize the equations which are used to update all actual control and knowledge means and 2nd-moment matrix partitions immediately following the execution of a guidance event. The actual estimation error means and 2nd-moment matrix partitions are updated using the following equations:

$$E\left[\tilde{x}_j^{+}\right] = E\left[\tilde{x}_j^{-}\right] - A \cdot E\left[\delta\Delta V_j^{-}\right] \tag{14}$$

$$E\left[\tilde{x}_{s_j}^{+}\right] = E\left[\tilde{x}_{s_j}^{-}\right] \tag{15}$$

$$P_{k_j}^{+} = P_{k_j}^{-} + A\, Q_j^{-}\, A^T - A \cdot E\left[\delta\Delta V_j^{-}\right] \cdot E\left[\tilde{x}_j^{-}\right]^T - E\left[\tilde{x}_j^{-}\right] \cdot E\left[\delta\Delta V_j^{-}\right]^T \cdot A^T \tag{16}$$

$$P_{s_{k_j}}^{+} = P_{s_{k_j}}^{-} \tag{17}$$

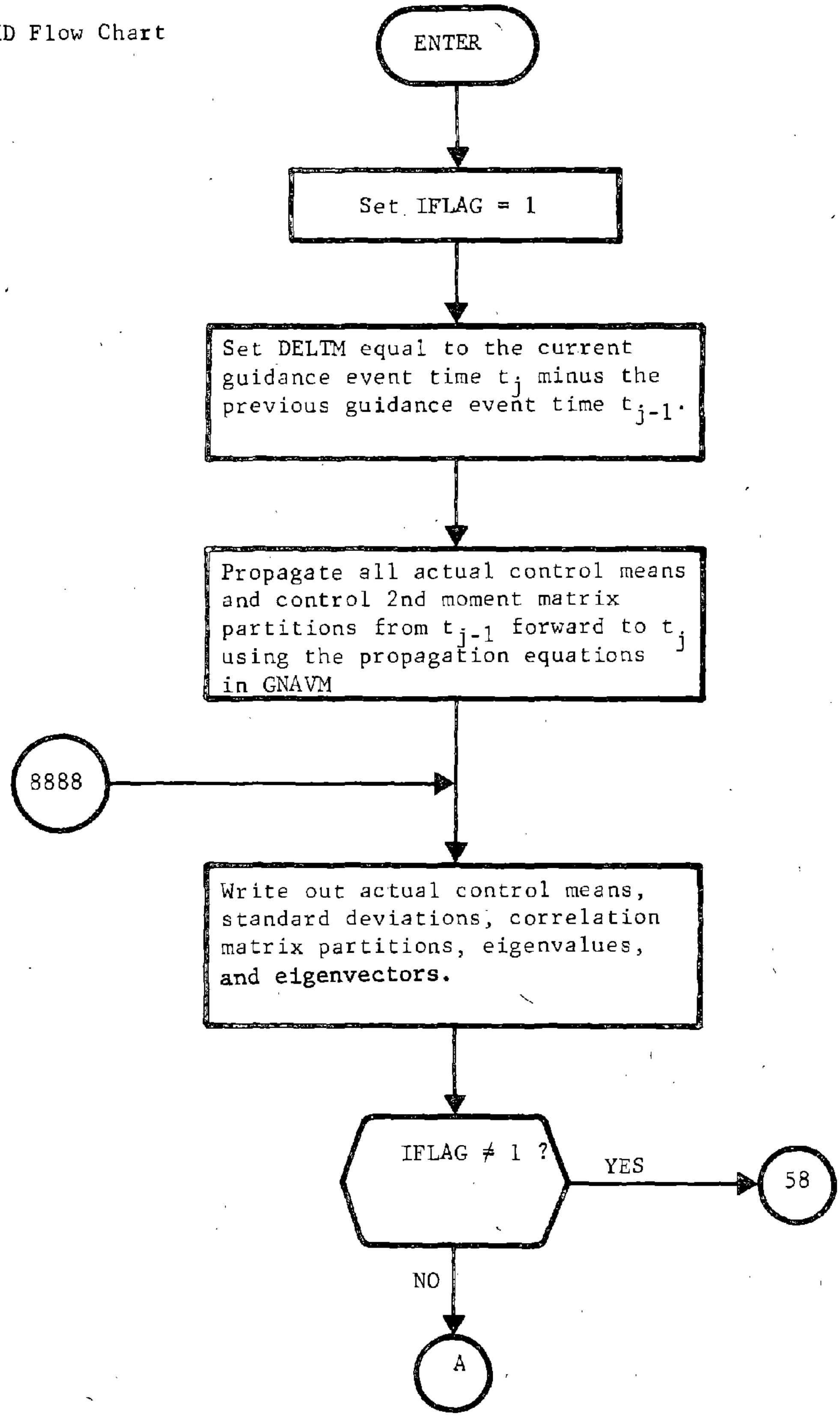
where $A = [0 \mid I]^T$. The actual deviation means are updated using the following equations:

$$E\left[x_j^{+}\right] = -E\left[\tilde{x}_j^{+}\right] \tag{18}$$

$$E\left[x_{s_j}^{+}\right] = -E\left[\tilde{x}_{s_j}^{+}\right] \tag{19}$$

The entire set of actual control 2nd-moment matrix partitions is updated by equating them to the corresponding actual knowledge 2nd-moment matrix partitions at t_j^{+}.

GENGID Flow Chart



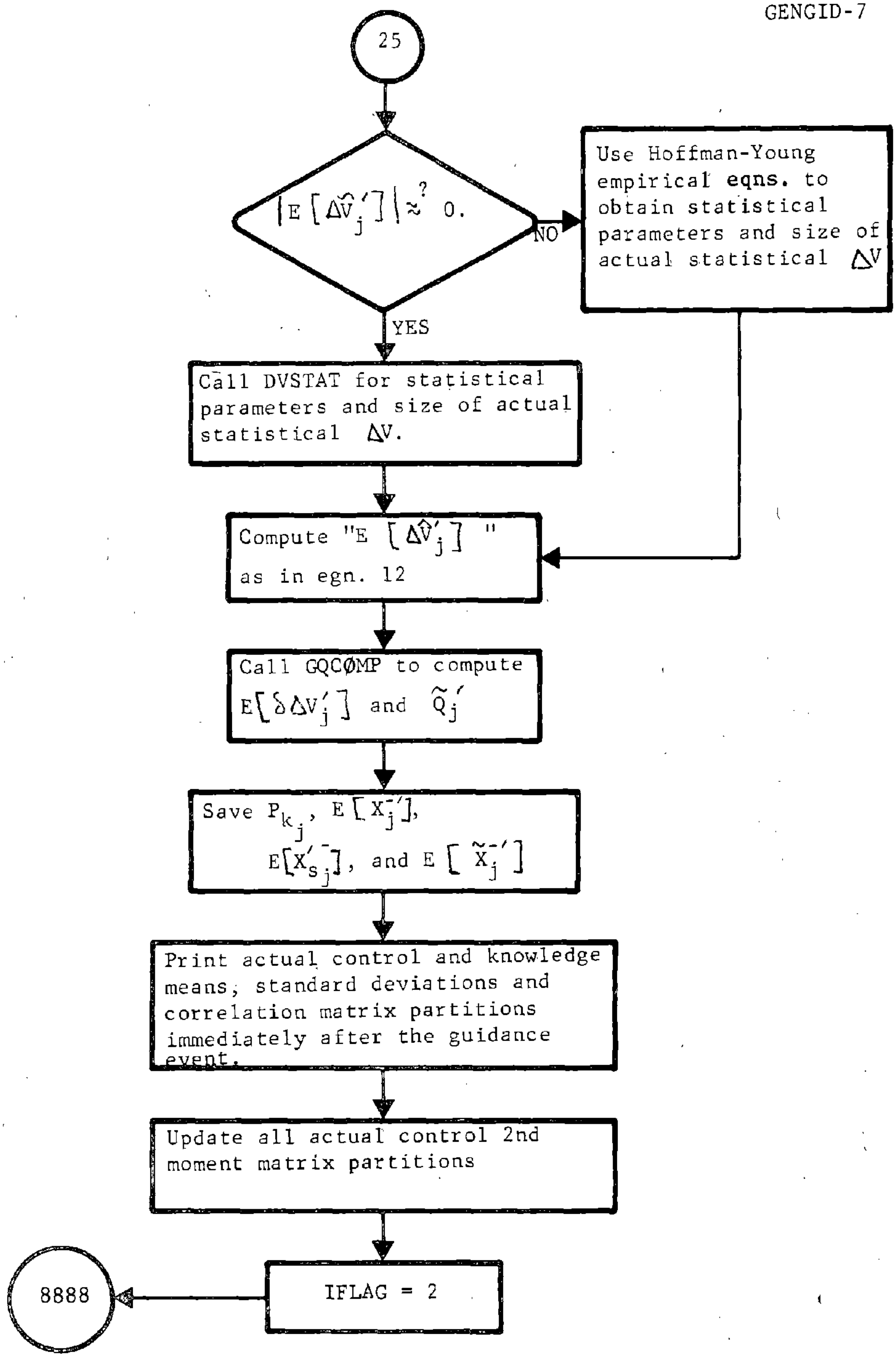
A

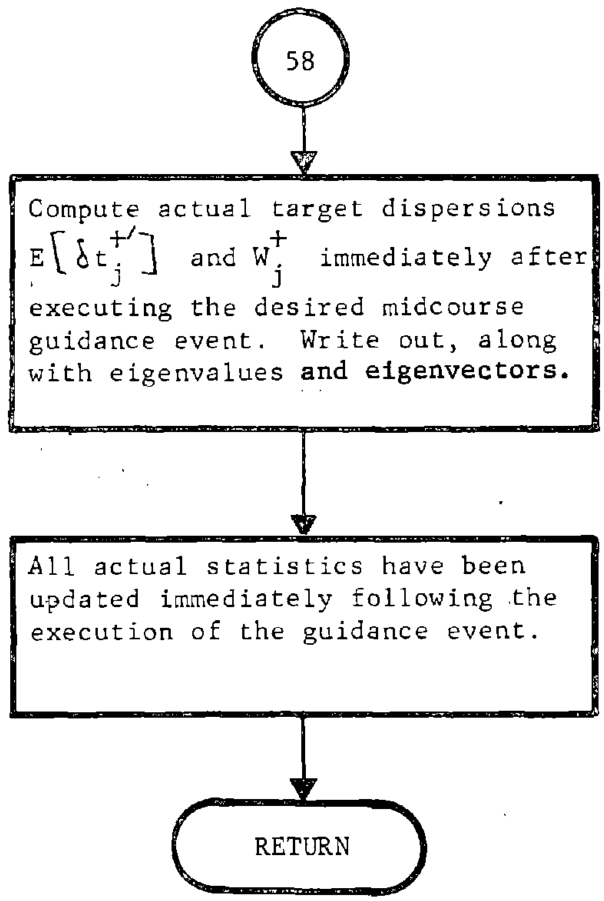
Compute actual target dispersions $E\left[\delta t_j^{-'}\right]$ and $W_j^{-'}$ immediately before executing the desired midcourse guidance event. Write out, along with eigenvalues and eigenvectors.

Compute acutal velocity connection 2nd moment matrix S_j'. Write out, along with eigenvalues and eigenvectors.

Compute $E\left[\Delta\hat{V}_j'\right]$

25


25
$|E[\Delta\hat{V}_j']| \overset{?}{\approx} 0.$
NO
Use Hoffman-Young empirical eqns. to obtain statistical parameters and size of actual statistical ΔV
YES
Call DVSTAT for statistical parameters and size of actual statistical ΔV.
Compute "$E[\Delta\hat{V}_j']$" as in egn. 12
Call GQCØMP to compute $E[\delta\Delta V_j']$ and $\tilde{Q}_j'$
Save P_{k_j}, $E[X_j^{-'}]$, $E[X_{s_j}'^{-}]$, and $E[\tilde{X}_j^{-'}]$
Print actual control and knowledge means, standard deviations and correlation matrix partitions immediately after the guidance event.
Update all actual control 2nd moment matrix partitions
IFLAG = 2
8888


58
Compute actual target dispersions $E\left[\delta t_j^{+/}\right]$ and W_j^{+} immediately after executing the desired midcourse guidance event. Write out, along with eigenvalues and eigenvectors.
All actual statistics have been updated immediately following the execution of the guidance event.
RETURN

SUBROUTINE GETCOW (ORBINT ENTRY POINT)

PURPOSE: TO GENERATE A STATE VECTOR AT A REQUESTED TIME BY INTERPOLATING DATA ON THE SEQUENTIAL ORBIT FILE

CALLING SEQUENCE: CALL GETCOW(NSEC,TREQ,IERR,X,STM)

ARGUMENTS:

```
         NSEC     I NUMBER OF DESIRED TRAJECTORY SECTION
         TREQ     I REQUEST TIME OF DATA (SEC)
         IERR     O ERROR FLAG
                    =1 NORMAL RETURN
                    =2 EOF DETECTED
                    =3 REQUESTED SECTION OUT OF RANGE
                    =4 REQUESTED TIME OUT OF RANGE
         X        O STATE VECTOR
         STM      O STATE PARTIALS
```

LOCAL SYMBOLS:

```
         IFRN       LOGICAL FILE NUMBER
```

SUBROUTINES REQUIRED:

```
         INTP
```

COMMON USED/COMPUTED:

```
         NEQ
```

COMMON COMPUTED:

```
         T          XVDD
         H          SV1
         XDD        SV2
         SX1        NSECTN
         SX2
```

SUBROUTINE GHA

PURPOSE: TO COMPUTE THE GREENWICH HOUR ANGLE AND THE UNIVERSAL TIME (IN DAYS) WHICH IS USED IN THE TRACKING MODULE TO ORIENT THE TRACKING STATIONS ON A SPERICAL ROTATING EARTH.

CALLING SEQUENCE: CALL GHA

ARGUMENTS: NONE

SUBROUTINES SUPPORTED: DATA1S DATA1

LOCAL SYMBOLS:

D	NUMBER OF DAYS IN TSTAR
EQMEG	EARTH ROTATION RATE
GH	GREENWICH HOUR ANGLE
ID	INTERMEDIATE VARIABLE
REFJD	JULIAN DATE OF JAN. 0, 1950
TFRAC	FRACTION OF DAY IN TSTAR
TSTAR	JULIAN DATE, EPOCH JAN. 0, 1950, OF INITIAL TRAJECTORY TIME

COMMON COMPUTED: UNIVT

COMMON USED: DATEJ EM13

GHA Analysis

Subroutine GHA computes the Greenwich hour angle in degrees and days at some epoch T* referenced to 1950 January 1^d0^h. Epoch T^* is computed from

$$T^* = J.D._0 + 2415020.0 - J.D._{REF}$$

where

$J.D._0$ = Julian date at launch time t_o referenced to 1900 January 0^d12^h.

$J.D._{REF}$ = Reference Julian date 2433282.5

= 1950 January 1^d0^h referenced to January 0^d12^h of the year 4713 B.C.

and 2415020.0 = 1900 January 0^d12^h referenced to January 0^d12^h of the year 4713 B.C.

Then T* is the Julian date at launch time t_o referenced to 1950 January 1^d0^h.

The Greenwich hour angle corresponding to T* is given by

$$GHA(T^*) = 100.0755426 + 0.985647346d + 2.9015 \times 10^{-13}\, d^2 + \omega t$$

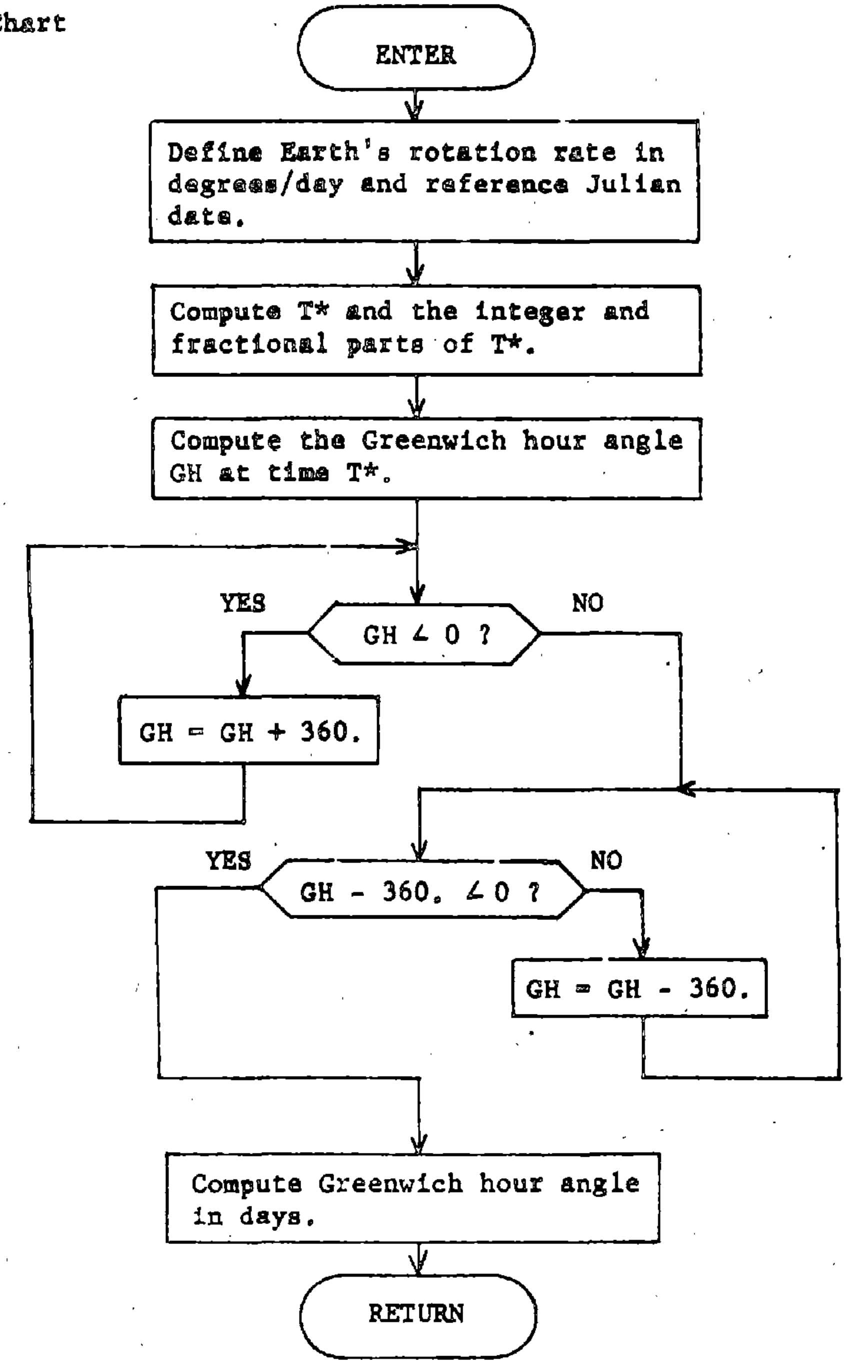
where $0 \leq GHA(T^*) < 360^\circ$.

and d = integer part of T*, t = fractional part of T*,

and ω = Earth's rotation rate is degrees/day.

The Greenwich hour angle im days is given by $\frac{GHA}{\omega}$.

GHA Flow Chart



SUBROUTINE GIDANS (PRELIM ENTRY POINT)

PURPOSE: DUMMY LINK WITH NON HALO ORBIT OPTIONS

CALLING SEQUENCE: CALL GIDANS

SUBROUTINE GNAVM

PURPOSE: TO PROPAGATE ASSUMED COVARIANCE MATRIX PARTITIONS P, CXXS,CXU,CXV,PS,CXSU,CXSV,OR ACTUAL SECOND MOMENT MATRIX PARTITIONS GP,GCXXS,GCXU,GCXV,GCXW,GPS,GCXSU,GCXSV, GCXSW FROM THE TIME OF THE LAST MEASUREMENT OR EVENT TO THE PRESENT TIME AND TO UPDATE THESE MATRIX PARTITIONS IF A MEASUREMENT IS TO BE PROCESSED

CALLING SEQUENCE: CALL GNAVM(NR,IFLAG1,ICODE,U0,V0,GCXW,GCXSW,
P,CXXS,CXU,CXV,PS,CXSU,CXSV,Q,R)

ARGUMENTS:

NR	I	NUMBER OF ROWS IN THE OBSERVATION MATRIX
IFLAG1	I	=1 FOR ASSUMED COVARIANCE PROCESSING =2 FOR ACTUAL SECOND MOMENT PROCESSING
ICODE	I	=0 FOR UPDATE =1 FOR PROPAGATION
U0	I	ACTUAL OR ASSUMED DYNAMIC CONSIDER PARAMETER 2ND MOMENT MATRIX
V0	I	ACTUAL OR ASSUMED MEASUREMENT CONSIDER PARAMETER 2ND MOMENT MATRIX
GCXW	I	ACTUAL POSITION-VELOCITY STATE / IGNORE PARAMETER 2ND MOMENT MATRIX
GCXSW	I	ACTUAL SOLVE-FOR PARAMETER / IGNORE PARAMETER 2ND MOMENT MATRIX
P	I	ACTUAL OR ASSUMED POSITION-VELOCITY 2ND MOMENT MATRIX 2ND MOMENT MATRIX
CXXS	I	ASSUMED OR ACTUAL POSITION-VELOCITY STATE / SOLVE-FOR PARAMETER 2ND MOMENT MATRIX
CXU	I	ASSUMED OR ACTUAL POSITION-VELOCITY STATE / DYNAMIC CONSIDER PARAMETER 2ND MOMENT MATRIX
CXV	I	ASSUMED OR ACTUAL POSITION-VELOCITY STATE / MEASUREMENT CONSIDER PARAMETER 2ND MOMENT MATRIX
PS	I	ASSUMED OR ACTUAL SOLVE-FOR PARAMETER COVARIANCE OR 2ND MOMENT MATRIX
CXSU	I	ASSUMED OR ACTUAL SOLVE-FOR PARAMETER / DYNAMIC CONSIDER PARAMETER 2ND MOMENT MATRIX

CXSV	I	ASSUMED OR ACTUAL SOLVE-FOR PARAMETER / MEASUREMENT CONSIDER PARAMETER 2ND MOMENT MATRIX
Q	I	ASSUMED OR ACTUAL DYNAMIC NOISE 2ND MOMENT MATRIX
R	I	ASSUMED OR ACTUAL MEASUREMENT NOISE 2ND MOMENT MATRIX

SUBROUTINES SUPPORTED: ERRANN SETEVN GUIDM PRED GENGID PROBE

SUBROUTINES REQUIRED: GAIN1 GAIN2

LOCAL SYMBOLS:

AKW	INTERMEDIATE ARRAY
DS	INTERMEDIATE ARRAY
ES	INTERMEDIATE ARRAY
FS	INTERMEDIATE ARRAY
IEND	NR-1
NDIM4S	NDIM4 VALUE STORAGE
N1	NDIM1-1
SUM	INTERMEDIATE VARIABLE
SW	INTERMEDIATE ARRAY

COMMON COMPUTED/USED:

AK	AL	AM	AN	CXSUP
CXSVP	CXUP	CXVP	CXXSP	G
GCUV	GCUW	GCVW	GCXSWP	GCXWP
HPHR	IGAIN	GW	H	HALF
JPR	ZERO	PP	PSP	S

COMMON USED:

NDIM1	NDIM2	NDIM3	NDIM4	PHI
TXU	TXW	TXXS		

GNAVM Analysis

Subroutine GNAVM propagates and updates (at a measurement) both assumed (or filter) covariance matrix partitions and actual 2nd moment matrix partitions. The equations programmed in GNAVM are independent of the filter algorithm employed to generate gain matrices.

The covariance and 2nd moment matrix partitions manipulated by GNAVM are defined as follows:

$$
\begin{aligned}
P &= E[\tilde{x}\,\tilde{x}^T] & P_s &= E\left[\tilde{x}_s\,\tilde{x}_s^T\right] \\
C_{xx_s} &= E\left[\tilde{x}\,\tilde{x}_s^T\right] & C_{x_s u} &= E[\tilde{x}_s\,\tilde{u}^T] \\
C_{xu} &= E[\tilde{x}\,\tilde{u}^T] & C_{x_s v} &= E[\tilde{x}_s\,\tilde{v}^T] \qquad (1) \\
C_{xv} &= E[\tilde{x}\,\tilde{v}^T] & C_{x_s w} &= E[\tilde{x}_s\,\tilde{w}^T] \quad . \\
C_{xw} &= E[\tilde{x}\,\tilde{w}^T]
\end{aligned}
$$

The following matrix partitions are used in GNAVM, but are not changed in GNAVM:

$$
\begin{aligned}
C_{uv} &= E[\tilde{u}\,\tilde{v}^T] \\
C_{uw} &= E[\tilde{u}\,\tilde{w}^T] \\
C_{vw} &= E[\tilde{v}\,\tilde{w}^T] \qquad (2) \\
U &= E[\tilde{u}\,\tilde{u}^T] \\
V &= E[\tilde{v}\,\tilde{v}^T] \\
W &= E[\tilde{w}\,\tilde{w}^T] \quad .
\end{aligned}
$$

In these definitions $\tilde{x}$, $\tilde{x}_s$, $\tilde{u}$, $\tilde{v}$, and $\tilde{w}$ represent, respectively, the estimation errors in position/velocity state, solve-for parameters, dynamic consider parameters, measurement consider parameters, and ignore parameters. Ignore parameters, of course, are not defined when assumed (or filter) covariance matrix partitions are being propagated or updated. Furthermore, the assumed C_{uv} has been set to zero.

The equations used to propagate covariances or 2nd moment matrices from time t_k to t_{k+1} are summarized:

$$P^-_{k+1} = \left(\Phi P^+_k + \theta_{xx_s} C^{+T}_{xx_{s_k}} + \theta_{xu} C^{+T}_{xu_k} + \theta_{xw} C^{+T}_{xw_k}\right)\Phi^T$$

$$+ C^-_{xx_{s_{k+1}}} \theta^T_{xx_s} + C^-_{xu_{k+1}} \theta^T_{xu} + C^-_{xw_{k+1}} \theta^T_{xw} + Q_{k+1} \tag{3}$$

$$C^-_{xx_{s_{k+1}}} = \Phi C^+_{xx_{s_k}} + \theta_{xx_s} P^+_{s_k} + \theta_{xu} C^{+T}_{x_s u_k} + \theta_{xw} C^{+T}_{x_s w_k} \tag{4}$$

$$C^-_{xu_{k+1}} = \Phi C^+_{xu_k} + \theta_{xx_s} C^+_{x_s u_k} + \theta_{xu} U_o + \theta_{xw} C^T_{uw_o} \tag{5}$$

$$C^-_{xv_{k+1}} = \Phi C^+_{xv_k} + \theta_{xx_s} C^+_{x_s v_k} + \theta_{xu} C_{uv_o} + \theta_{xw} C^T_{vw_o} \tag{6}$$

$$C^-_{xw_{k+1}} = \Phi C^+_{xw_k} + \theta_{xx_s} C^+_{x_s w_k} + \theta_{xu} C_{uw_o} + \theta_{xw} W_o \tag{7}$$

$$P^-_{s_{k+1}} = P^+_{s_k} \tag{8}$$

$$C^-_{x_s u_{k+1}} = C^+_{x_s u_k} \tag{9}$$

$$C^-_{x_s v_{k+1}} = C^+_{x_s v_k} \quad . \tag{10}$$

$$C^-_{x_s w_{k+1}} = C^+_{x_s w_k} \quad . \tag{11}$$

In these equations $(\ )^-$ indicates immediately prior to processing a measurement; $(\ )^+$, immediately after. The state transition matrices over the interval $[t_k, t_{k+1}]$ are indicated by Φ, θ_{xx_s}, θ_{xu}, and θ_{xw}. The dynamic noise covariance or 2nd moment matrix is denoted by Q_{k+1}.

Before covariance (or 2nd moment) matrix partitions can be updated at a measurement, the measurement residual covariance (or 2nd moment) matrix, defined by

$$J_{k+1} = E\left[\varepsilon_{k+1}\ \varepsilon^T_{k+1}\right] \tag{12}$$

must be computed. The required equations are summarized

$$J_{k+1} = HA_{k+1} + MB_{k+1} + GD_{k+1} + LE_{k+1} + NF_{k+1} + R_{k+1} \tag{13}$$

$$A_{k+1} = P^-_{k+1} H^T + C^-_{xx_{s_{k+1}}} M^T + C^-_{xu_{k+1}} G^T + C^-_{xv_{k+1}} L^T + C^-_{xw_{k+1}} N^T \tag{14}$$

$$B_{k+1} = P^-_{s_{k+1}} M^T + C^{-T}_{xx_{s_{k+1}}} H^T + C^-_{x_s u_{k+1}} G^T + C^-_{x_s v_{k+1}} L^T + C^-_{x_s w_{k+1}} N^T \tag{15}$$

$$D_{k+1} = C^{-T}_{xu_{k+1}} H^T + C^{-T}_{x_s u_{k+1}} M^T + U_o G^T + C^-_{uw_o} N^T + C^-_{uv_o} L^T \tag{16}$$

$$E_{k+1} = C^{-T}_{xv_{k+1}} H^T + C^{-T}_{x_s v_{k+1}} M^T + C^-_{vw_o} N^T + V_o L^T + C^{-T}_{uv_o} G^T \tag{17}$$

$$F_{k+1} = W_o N^T + C^{-T}_{xw_{k+1}} H^T + C^{-T}_{x_s w_{k+1}} M^T + C^{-T}_{vw_o} L^T + C^{-T}_{uw_o} G^T \quad . \tag{18}$$

In these equations H, M, G, L, and N represent observation matrix partitions, and R_{k+1} represents the measurement noise covariance (or 2nd moment) matrix.

Gain matrices K_{k+1} and S_{k+1} are also required before covariance (or 2nd moment) matrix partitions can be updated. These are not computed in GNAVM but are obtained by calling either subroutine GAIN1 or GAIN2, depending on which recursive estimation algorithm is desired.

With J_{k+1}, K_{k+1}, and S_{k+1} available, the following equations are used in the updating process:

$$P^+_{k+1} = P^-_{k+1} - K_{k+1} A^T - AK^T_{k+1} + K_{k+1} J_{k+1} K^T_{k+1} \tag{19}$$

$$C^+_{xx_{s_{k+1}}} = C^-_{xx_{s_{k+1}}} - K_{k+1} B^T - AS^T_{k+1} + K_{k+1} J_{k+1} S^T_{k+1} \tag{20}$$

$$C^+_{xu_{k+1}} = C^-_{xu_{k+1}} - K_{k+1} D^T \tag{21}$$

$$C^+_{xv_{k+1}} = C^-_{xv_{k+1}} - K_{k+1} E^T \tag{22}$$

$$C^+_{xw_{k+1}} = C^-_{xw_{k+1}} - K_{k+1} F^T \tag{23}$$

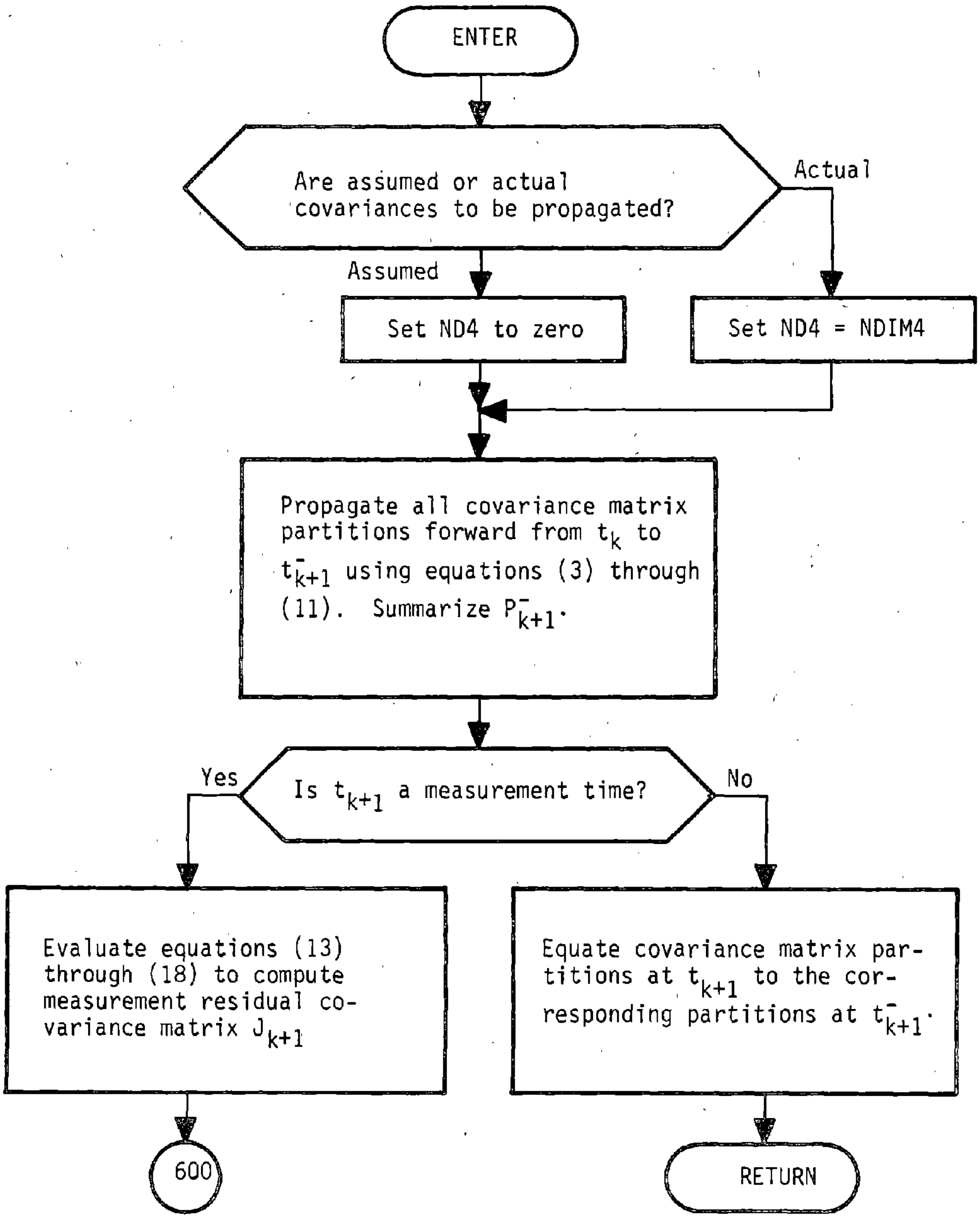
$$P^+_{s_{k+1}} = P^-_{s_{k+1}} - S_{k+1} B^T - BS^T_{k+1} + S_{k+1} J_{k+1} S^T_{k+1} \tag{24}$$

$$C^+_{x_s u_{k+1}} = C^-_{x_s u_{k+1}} - S_{k+1} D^T \tag{25}$$

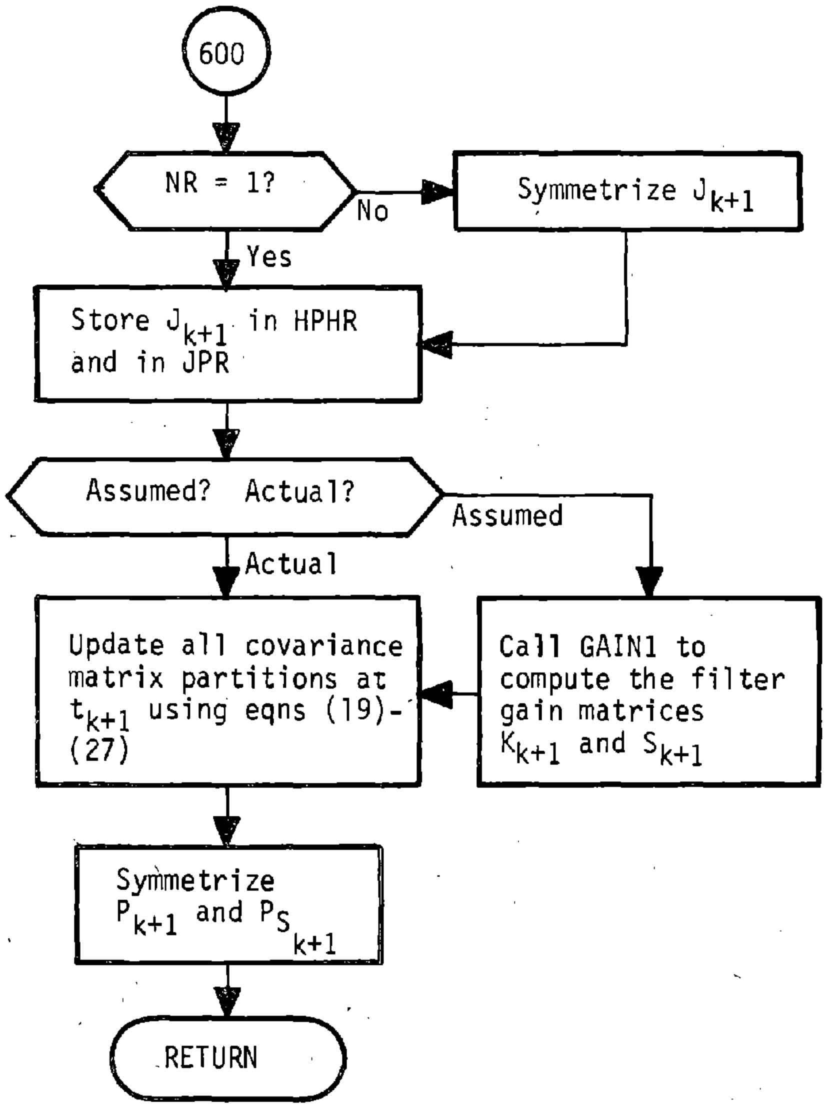
$$C^{+}_{x_s v_{k+1}} = C^{-}_{x_s v_{k+1}} - S_{k+1} E^T \tag{26}$$

$$C^{+}_{x_s w_{k+1}} = C^{-}_{x_s w_{k+1}} - S_{k+1} F^T \quad . \tag{27}$$

It should be noted that propagation equations (3) through (11) are also used to propagate both assumed control covariance and actual 2nd moment matrix partitions over the time interval separating two successive guidance events. The update equations, of course, are not used in this situation.



GNAVM Flow Chart


600
NR = 1?
No
Symmetrize J_{k+1}
Yes
Store J_{k+1} in HPHR and in JPR
Assumed? Actual?
Assumed
Actual
Update all covariance matrix partitions at t_{k+1} using eqns (19)-(27)
Call GAIN1 to compute the filter gain matrices K_{k+1} and S_{k+1}
Symmetrize P_{k+1} and $P_{S_{k+1}}$
RETURN

SUBROUTINE GPRINT

PURPOSE: TO PRINT ACTUAL ESTIMATION ERROR STATISTICS

CALLING SEQUENCE: CALL GPRINT(IFLAG,TIMM)

ARGUMENTS: IFLAG I =3 PRINT ACTUAL STATISTICS AT A GUIDANCE EVENT
=10 PRINT ACTUAL ESTIMATION ERROR STATISTICS
=2 PRINT ACTUAL ESTIMATION ERROR STATISTICS AT A PREDICTION EVENT

TIMM I TIME TO BE PRINTED

SUBROUTINES SUPPORTED: PRED ERRANN SETEVN

SUBROUTINES REQUIRED: MOMENT

LOCAL SYMBOLS:

A	HOLLERITH WORD -AFTER-
B	HOLLERITH WORD -BEFORE-
DUM	INTERMEDIATE VECTOR
EXSTSV	TEMPORARY STORAGE FOR EXST
EXTSV	TEMPORARY STORAGE FOR EXT
ROW	INTERMEDIATE VECTOR
ZZ	INTERMEDIATE VARIABLE

COMMON USED:

CXSUP	CXSVP	CXUP	CXVP	CXXSP
EMRES	EU	EV	EW	EXI
EXSI	EXST	EXSTP	EXTP	GCXSU
GCXSV	GCXSW	GCXSWP	GCXU	GCXV
GCXW	GCXWP	GCXXS	GP	GPS
GU	GV	GW	JPR	NDIM1
NDIM2	NDIM3	NDIM4	NR	PP
PSP	RPR	TRTM2	XIG	XLAB
XSL	XU	XV		

```
SUBROUTINE    GQCOMP

PURPOSE: TO COMPUTE ACTUAL EXECUTION ERROR STATISTICS

CALLING SEQUENCE:  CALL GQCOMP(VV,EE,EEE,EV,Q)

ARGUMENTS:  VV    I ACTUAL COMMANDED VELOCITY CORRECTION

            EE    I MEANS OF ACTUAL EXECUTION ERRORS

            EEE   I 2ND MOMENTS OF ACTUAL EXECUTION ERRORS

            EV    O EXPECTED VALUE OF ACTUAL EXECUTION ERROR

            Q     O ACTUAL EXECUTION ERROR 2ND MOMENT MATRIX

SUBROUTINES SUPPORTED:  GENGID

LOCAL SYMBOLS:  FACTR   INTERMEDIATE VARIABLE

                RHOP    MAGNITUDE OF VV VECTOR

                RHOP2   RHOP**2

                V1      V(1)**2

                V2      V(2)**2

                V3      V(3)**2

                V4      V1*V2*V3

                XI      INTERMEDIATE VARIABLE

                XMUP    INTERMEDIATE VARIABLE

                ZETA    INTERMEDIATE VARIABLE
```

GQCØMP Analysis

Subroutine GQCØMP computes the actual execution error mean and 2nd moment matrix for use in the generalized covariance analysis of a guidance event. The actual execution error $\delta\Delta V_j'$ is assumed to have the form

$$\delta\Delta V_j' = k' \; \Delta\hat{V}_j' + s' \; \frac{\Delta\hat{V}_j'}{|\Delta\hat{V}_j'|} + \delta\Delta V'_{pointing} \tag{1}$$

where k' denotes the actual proportionality error; s', the actual resolution error; $\delta\Delta V'_{pointing}$, the actual pointing error; and $\Delta\hat{V}_j'$, the actual commanded velocity correction.

The means of the three ecliptic components of $\delta\Delta V_j'$ are given as:

$$E[\delta\Delta V_x'] = \left(\bar{k}' + \frac{\bar{s}'}{\rho'}\right) \Delta\hat{V}_x' + \frac{\rho'\Delta\hat{V}_y' \; \overline{\delta\alpha}' + \Delta\hat{V}_x' \; \Delta\hat{V}_z' \; \overline{\delta\beta}'}{\mu'} \tag{2}$$

$$E[\delta\Delta V_y'] = \left(\bar{k}' + \frac{\bar{s}'}{\rho'}\right) \Delta\hat{V}_y' + \frac{\Delta\hat{V}_y' \; \Delta\hat{V}_z' \; \overline{\delta\beta}' - \rho'\Delta\hat{V}_x' \; \overline{\delta\alpha}'}{\mu'} \tag{3}$$

$$E[\delta\Delta V_z'] = \left(\bar{k}' + \frac{\bar{s}'}{\rho'}\right) \Delta\hat{V}_z' - \mu' \; \overline{\delta\beta}' \tag{4}$$

where $\rho' = |\Delta\hat{V}'|$, $\mu' = [\Delta\hat{V}_x'^2 + \Delta\hat{V}_y'^2]^{\frac{1}{2}}$, and $\delta\alpha'$ and $\delta\beta'$ are the actual pointing angle errors, and both E( ) and ($\bar{}$) indicate mean values.

The actual execution error 2nd moment matrix is defined by

$$\tilde{Q}_j' = E\left[\delta\Delta V_j' \;\; \delta\Delta V_j'^T\right] \quad . \tag{5}$$

the elements Q_{ik}' of matrix Q_j' are given as:

$$\tilde{Q}_{11}' = \xi' \; \Delta\hat{V}_x'^2 + \frac{1}{\mu'^2}\left(\rho'^2\Delta\hat{V}_y'^2 \; \overline{\delta\alpha' \; \delta\alpha'} + \Delta\hat{V}_x'^2 \; \Delta\hat{V}_z'^2 \; \overline{\delta\beta' \; \delta\beta'} + \right.$$

$$\left. 2\rho' \; \Delta\hat{V}_x' \; \Delta\hat{V}_y' \; \Delta\hat{V}_z' \; \overline{\delta\alpha}' \; \overline{\delta\beta}'\right) + \frac{2\Delta\hat{V}_x'}{\mu'} \zeta' \left(\rho\Delta\hat{V}_y' \; \overline{\delta\alpha}' + \Delta\hat{V}_x' \; \Delta\hat{V}_z' \; \overline{\delta\beta}'\right) \tag{5}$$

$$\tilde{Q}'_{22} = \xi' \, \Delta\hat{V}'^2_y + \frac{1}{\mu'^2}\left(\Delta\hat{V}'^2_y \, \Delta\hat{V}'^2_z \, \overline{\delta\beta' \, \delta\beta'} + \rho'^2 \, \Delta\hat{V}'^2_x \, \overline{\delta\alpha' \, \delta\alpha'} - \right.$$

$$\left. 2\rho' \, \Delta\hat{V}'_x \, \Delta\hat{V}'_y \, \Delta\hat{V}'_z \quad \overline{\delta\alpha'} \, \overline{\delta\beta'}\right) + \frac{2\Delta\hat{V}'_y}{\mu'} \zeta' \left(\Delta\hat{V}'_y \, \Delta\hat{V}'_z \quad \overline{\delta\beta'} - \rho' \, \Delta\hat{V}'_x \, \overline{\delta\alpha'}\right) \quad (6)$$

$$\tilde{Q}'_{33} = \xi' \, \Delta\hat{V}'^2_z + \mu'^2 \, \overline{\delta\beta' \, \delta\beta'} - 2\Delta\hat{V}'_z \, \mu' \, \zeta' \, \overline{\delta\beta'} \quad (7)$$

$$\tilde{Q}'_{12} = \tilde{Q}'_{21} = \xi' \, \Delta\hat{V}'_x \, \Delta\hat{V}'_y + \frac{\zeta'}{\mu'}\left|2\Delta\hat{V}'_x \, \Delta\hat{V}'_y \, \Delta\hat{V}'_z \quad \overline{\delta\beta'} - \rho'\left(\Delta\hat{V}'^2_x - \Delta\hat{V}'^2_y\right)\overline{\delta\alpha'}\right| +$$

$$\frac{1}{\mu'^2}\left|- \rho'^2 \, \Delta\hat{V}'_x \, \Delta\hat{V}'_y \, \overline{\delta\alpha' \, \delta\alpha'} \quad + \rho' \, \Delta\hat{V}'_z \left(\Delta\hat{V}'^2_y - \Delta\hat{V}'^2_x\right) \overline{\delta\alpha'} \, \overline{\delta\beta'} + \right.$$

$$\left. \Delta\hat{V}'_x \, \Delta\hat{V}'_y \, \Delta V'^2_z \, \overline{\delta\beta' \, \delta\beta'}\right| \quad (8)$$

$$\tilde{Q}'_{13} = \tilde{Q}'_{31} = \xi' \, \Delta\hat{V}'_x \, \Delta\hat{V}'_z \quad + \zeta'\left[\frac{\Delta\hat{V}'_z}{\mu'}\left(\rho'\Delta\hat{V}'_y \, \overline{\delta\alpha'} + \Delta\hat{V}'_x \, \Delta\hat{V}'_z \, \overline{\delta\beta'}\right) - \mu' \, \Delta\hat{V}'_x \, \overline{\delta\beta'}\right]$$

$$- \rho'\Delta\hat{V}'_y \, \overline{\delta\alpha'} \, \overline{\delta\beta'} - \Delta\hat{V}'_x \, \Delta\hat{V}'_z \, \overline{\delta\beta' \, \delta\beta'} \quad (9)$$

$$\tilde{Q}'_{23} = \tilde{Q}'_{32} = \xi' \, \Delta\hat{V}'_y \, \Delta\hat{V}'_z + \zeta'\left[\frac{\Delta\hat{V}'_z}{\mu'}\left(\Delta\hat{V}'_y \, \Delta\hat{V}'_z \quad \overline{\delta\beta'} - \rho'\Delta\hat{V}'_x \, \overline{\delta\alpha'}\right) - \mu' \, \Delta\hat{V}'_y \, \overline{\delta\beta'}\right]$$

$$+ \rho'\Delta\hat{V}'_x \, \overline{\delta\alpha'} \, \overline{\delta\beta'} - \Delta\hat{V}'_y \, \Delta\hat{V}'_z \, \overline{\delta\beta' \, \delta\beta'} \quad (10)$$

where

$$\xi' = \overline{k' \, k'} + \frac{2}{\rho'} \overline{k'} \, \overline{s'} + \frac{\overline{s' \, s'}}{\rho'^2} \quad (11)$$

and

$$\zeta' = \overline{k'} + \frac{\overline{s'}}{\rho'} \quad (12)$$

SUBROUTINE GUID

PURPOSE: COMPUTE THE GUIDANCE MATRIX, THE VARIATION MATRIX, AND THE TARGET CONDITION (JUST BEFORE) COVARIANCE MATRIX AT A MIDCOURSE GUIDANCE EVENT

CALLING SEQUENCE: CALL GUID(RF,IGP,TEXN,GA,ADA)

ARGUMENTS:

RF	STATE AT TIME OF EVENT
IGP	GUIDANCE POLICY (=1 FOR FTA, =2 FOR VTA)
TEVN	TIME OF EVENT
GA	GUIDANCE MATRIX
ADA	VARIATION MATRIX

SUBROUTINE SUPPORTED: GUIDM

SUBROUTINES REQUIRED: CSTART EIGHY MATIN NTM PSIM SHIFT STMPR ZERMAT

LOCAL SYMBOLS:

ALF	ARRAY OF HOLLERITH CONSTANTS FOR PRINT
DUM1	INTERMEDIATE 2X2 MATRIX
EGVL	VECTOR OF EIGENVALUES
IERR	ERROR FLAG RETURNED BY FILE READER
PHI3	INTERMEDIATE 3X3 MATRIX
ROT	ROTATION MATRIX FOR VTA COMPUTATIONS
ROW	TEMPORARY STORAGE VECTOR
SQP	TEMPORARY STORAGE VECTOR
YPM	INTERMEDIATE VALUE
ZPM	INTERMEDIATE VALUE
ZTM	INTERMEDIATE VALUE

COMMON COMPUTED/USED: DELTM PHI TRTM1

COMMON USED: FISAVE FISAVE

GUID Analysis

Subroutine GUID is called at a midcourse guidance event at t_j in the error analysis mode to compute three primary quantities for the selected midcourse guidance policy. These three quantities are the variation matrix η_j, the target condition covariance matrix prior to the velocity correction W_j^-, and the guidance matric Γ_j. Two midcourse guidance policies are available: fixed-time-of-arrival (FTA), and variable-time-of-arrival (VTA). Both are linear impulsive guidance policies having form

$$\Delta V_j = \Gamma_j \ \delta X_j$$

where ΔV_j is the commanded velocity correction, and δX_j is the estimate of the spacecraft position/velocity deviation from the targeted nominal. The relevant equations for each guidance policy will be summarized below.

The variation matrix η_j for FTA guidance relates deviations in spacecraft state at t_j to position deviations at the final time t_F, and is given by

$$\eta_j = \left[\phi_1 \;\vdots\; \phi_2\right]$$

where $\left[\phi_1 \;\vdots\; \phi_2\right]$ is the upper half of the state transition matrix $\Phi(t_F, t_j)$. The guidance matrix for FTA guidance is given by

$$\Gamma_j = \left[-\phi_2^{-1}\,\phi_1 \;\vdots\; -1\right]$$

The variation matrix for VTA guidance relates deviations in state at t_j to deviations from the nominal normal to the impulsive insertion velocity vector. Consequently, the state deviations at the nominal target time, t_F, are rotated to a coordinate system whose z-axis is along the input delta-V, REXV, and only the upper 2 x 6 partition of the rotated $\Phi(t_F, t_j)$ matrix becomes η

$$\eta_{VTA} = \left[A \;\vdots\; B\right]$$

where

$$A = \left[R\phi_1\right]_{2x3} \qquad B = \left[R\phi_2\right]_{2x3}$$

Next the guidance matrix is obtained by

$$\Gamma_{VTA} = \left[-B^T(BB^T)^{-1} A \;\vdots\; -B^T(BB^T)^{-1} B \right]$$

$$= -B^T(BBT)^{-1} \eta_{VTA}$$

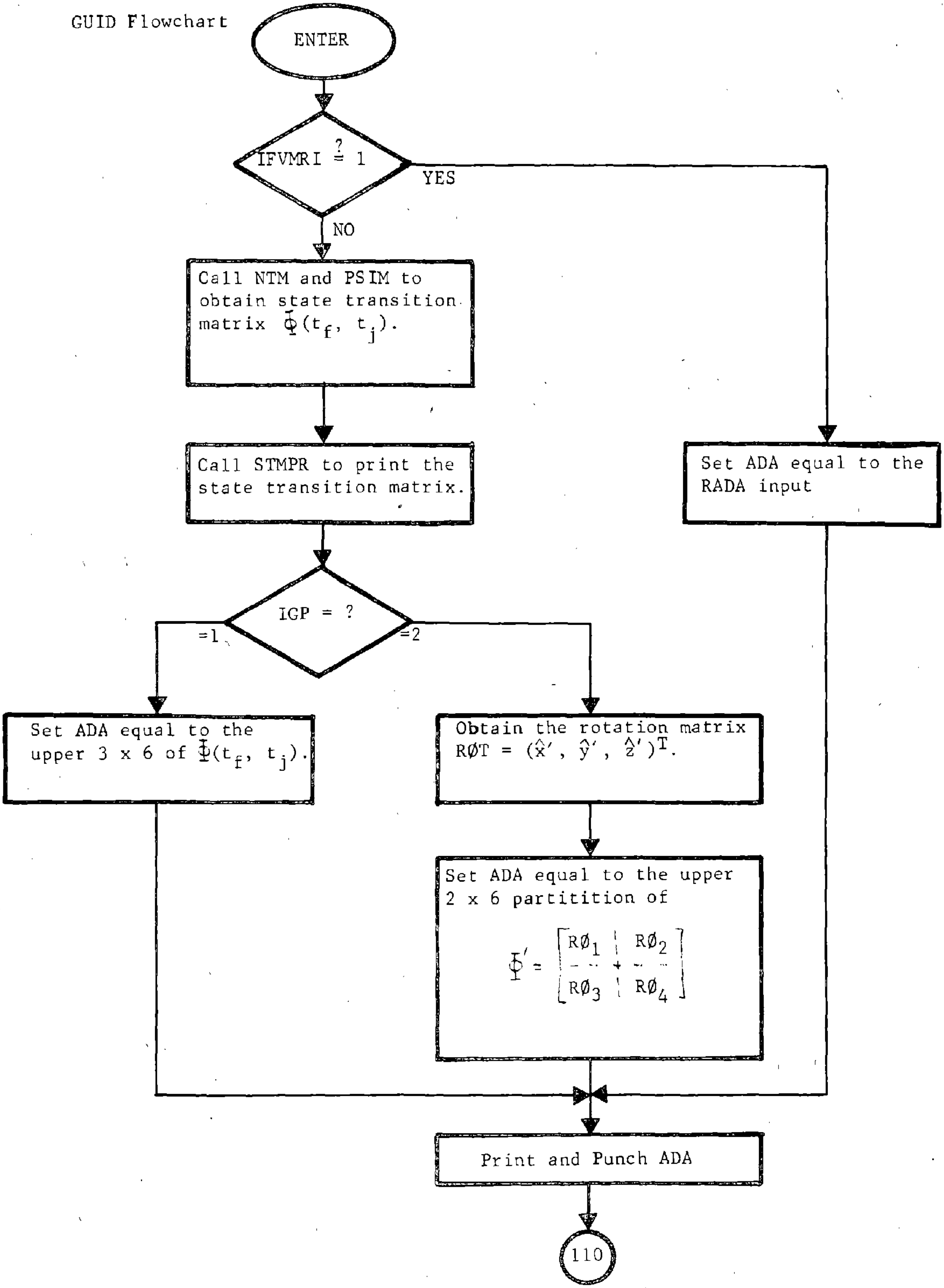
Whichever guidance policy is used to obtain η_j and Γ_j, the target condition covariance matrix is computed as

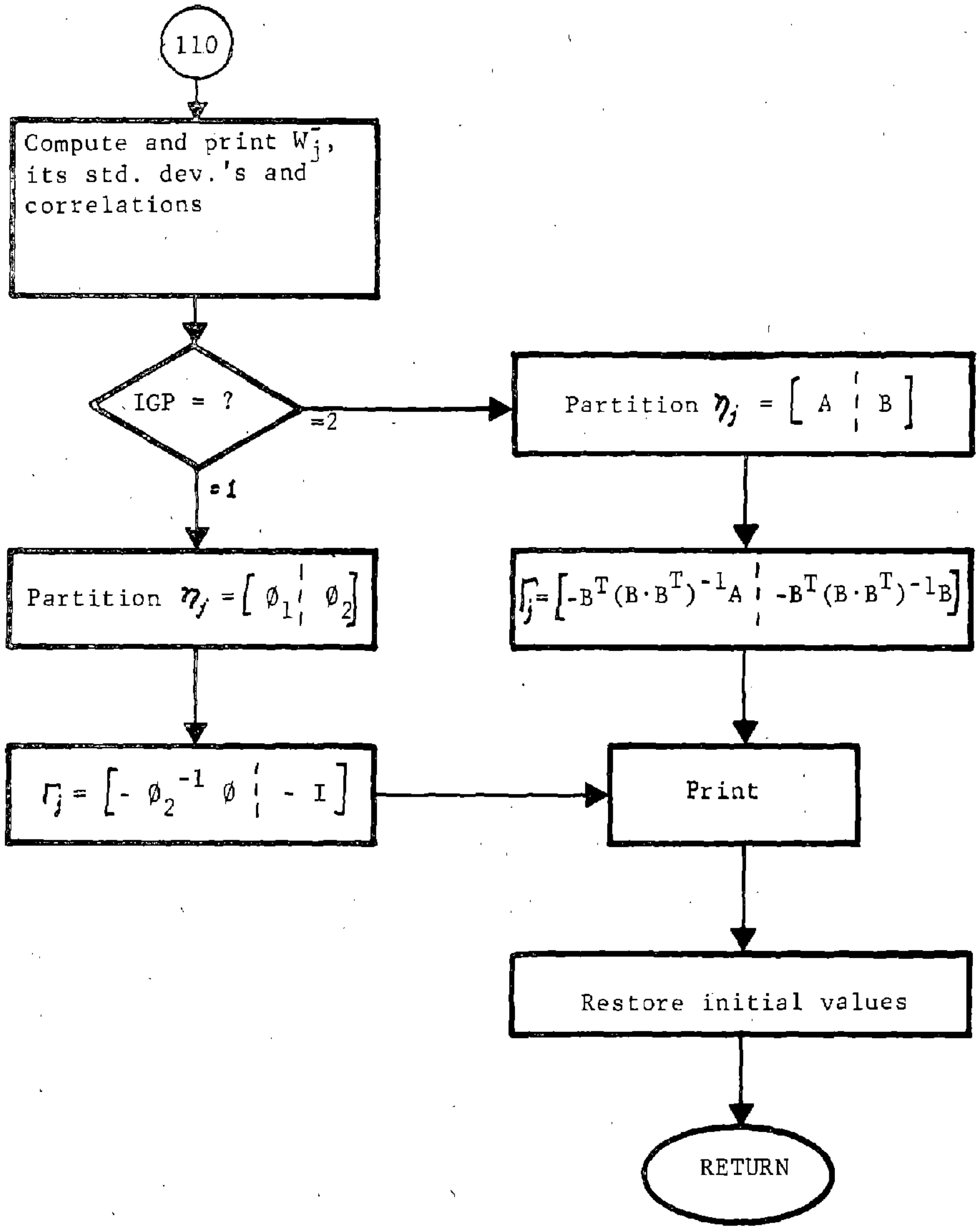
$$W_J^- = \eta_j P_{c_j}^- \eta_j^T$$

Where $P_{c_j}^-$ is the control covariance matrix immediately prior to the to the guidance event.

Finally, saved values are restored and the trajectory file is re-initialized for use by the file reader when called from other subroutines.

GUID Flowchart




110
Compute and print W_j^-, its std. dev.'s and correlations
IGP = ?
=2
=1
Partition $\eta_j = [A \mid B]$
Partition $\eta_j = [\phi_1 \mid \phi_2]$
$\Gamma_j = [-B^T(B \cdot B^T)^{-1}A \mid -B^T(B \cdot B^T)^{-1}B]$
$\Gamma_j = [-\phi_2^{-1}\phi \mid -I]$
Print
Restore initial values
RETURN

SUBROUTINE GUIDM

PURPOSE CONTROL EXECUTION OF A GUIDANCE EVENT IN THE ERROR ANALYSIS PROGRAM

CALLING SEQUENCE: CALL GUIDM

SUBROUTINES SUPPORTED: ERRANN

SUBROUTINES REQUIRED: CORREL DYNO GUID EIGHY JACOBI GNAVM DVSTAT NTM PSIM STMPR CSTART SAVMAT

LOCAL SYMBOLS:

ADA	VARIATION MATRIX
AMAX	INTERMEDIATE VARIABLE USED TO FIND MAXIMUM EIGENVALUE OF VELOCITY CORRECTION COVARIANCE MATRIX (S MATRIX)
CXSU1	STORAGE FOR CXSU KNOWLEDGE COVARIANCE
CXSV1	STORAGE FOR CXSV KNOWLEDGE COVARIANCE
CXU1	STORAGE FOR CXU KNOWLEDGE COVARIANCE
CXV1	STORAGE FOR CXV KNOWLEDGE COVARIANCE
CXXS1	STORAGE FOR CXXS KNOWLEDGE COVARIANCE
DUM1	INTERMEDIATE VARIABLE
DUM	VECTOR SUM OF UPDATE AND STATISTICAL VELOCITY CORRECTIONS
EGM	MAXIMUM EIGENVALUE OF S MATRIX
EGVCT	ARRAY OF EIGENVECTORS
EGVL	ARRAY OF EIGENVALUES
EXEC	EXECUTION ERROR COVARIANCE MATRIX
EXV	EXPECTED VALUE OF VELOCITY CORRECTION
GA	GUIDANCE MATRIX
GAP	INTERMEDIATE ARRAY EQUAL TO GA TIMES P
ICODE	INTERNAL CONTROL FLAG
ICODE2	INTERNAL CONTROL FLAG
IGP	MIDCOURSE GUIDANCE POLICY CODE

ISPHC	TEMPORARY STORAGE FOR ISPH
MAP	INDEX OF MAXIMUM EIGENVALUE OF S
OUT	SPACECRAFT VELOCITY RELATIVE TO TARGET PLANET IN PLANETO-CENTRIC EQUATORIAL COORDINATES
P2	STORAGE FOR P CONTROL COVARIANCE
P1	STORAGE FOR P KNOWLEDGE COVARIANCE
RF	NOMINAL TRAJECTORY STATE AT GUIDANCE EVENT
RHO	MAGNITUDE OF STATISTICAL DELTA-V
ROW	INTERMEDIATE VECTOR
SQP	INTERMEDIATE VECTOR
TRS	TRACE OF S MATRIX
VEIG	MATRIX TO BE DIAGONALIZED
Z	INTERMEDIATE ARRAY

COMMON COMPUTED/USED:		CXSUG	CXSU	CXSVG	CXSV	CXUG	
		CXU	CXVG	CXV	CXXSG	CXXS	
		ISPH	NGE	PG	PSG	PS	
		P	TG	XG			
COMMON COMPUTED:		DELTM	TRTM1	XI			
COMMON USED:	FOP	FOV	ICDT3	NDIM1	NDIM2	NDIM3	ONE
	Q	SKALE	SIGALP	SIGBET	SIGPRO	SIGRES	TWO
	U0	V0	XF	ZERO			

GUIDM Analysis

Subroutine GUIDM is the executive guidance subroutine in the error analysis program. In addition to controlling the computational flow for all types of guidance events, GUIDM also performs many of the required guidance computations itself.

Before considering each type of guidance event, the treatment of a general guidance event will be discussed. Let t_j be the time at which the guidance event occurs. Before any guidance event can be executed, the targeted nominal state $\bar{X}_j$, knowledge covariance P_{K_j}, and control covariance P_{c_j} must all be available, where $(\ )^-$ indicates values immediately before the event. The first two quantities are available prior to entering GUIDM. However, GUIDM controls the propagation of the control covariance over the interval $[t_{j-1}, t_j]$, where t_{j-1} denotes the time of the previous guidance event.

The next step in the treatment of a general guidance event is concerned with the computation of the effective velocity correction and the execution error covariance. In the error analysis program, only a statistical velocity correction can be computed. The effective velocity correction ΔV_j is then used to compute the execution error covariance matrix $\tilde{Q}_j$. A summary of the execution error model and the equations used to compute $\tilde{Q}_j$ can be found in the subroutine GQCØMP analysis section.

The last step is concerned with the updating of required quantities prior to returning to the basic cycle. An assumption underlying the modeled guidance process is that the targeted nominal remains unchanged at a guidance event.

The knowledge covariance is updated using the equation

$$P_{K_j}^{+} = P_{K_j}^{-} + \left[\begin{array}{c|c} 0 & 0 \\ \hline 0 & \tilde{Q}_j \end{array}\right]$$

if an impulsive thrust model is assumed.

In the impulsive case, the control covariance is updated simply by setting

$$P_{c_j}^{+} = P_{K_j}^{+} \quad .$$

This equation is a direct consequence of the assumption that the targeted nominal state is always updated at a guidance event.

Each specific type of guidance event involves the computation of other quantities not discussed above. These will be covered in the following discussion of specific guidance events.

1. Midcourse Guidance

Linear midcourse guidance policies have form

$$\Delta V_{N_j} = \Gamma_j \; \delta X_j$$

where the subscript N indicates that this is the velocity correction required to null out deviations from the nominal target state. This notation is used to differentiate between this type of velocity correction and velocity corrections required to achieve final insertion. Linear midcourse guidance policies are discussed in more detail in the subroutine GUID analysis section.

Subroutine GUIDM calls GUID to compute the guidance matrix, Γ_j, and the target condition covariance immediately prior to the guidance event, W_j, and then uses Γ_j to compute the velocity correction covariance S_j, which is defined as:

$$S_j = E \left[\Delta V_{N_j} \quad \Delta V_{N_j}^{\;T} \right] ,$$

and is given by the equation

$$S_j = \Gamma_j \left(P_{c_j}^- - s \, P_{K_j}^- \right) \Gamma_j^T .$$

where s is a (real) scalar, input by the analyst; generally, $0. \leq s \leq 1$. This equation assumes that an optimal estimation algorithm is employed in the navigation process, since the derivation of this equation requires the orthogonality of the estimate and the estimation error.

In the error analysis program ΔV_{N_j} is never available since no estimates δX_j are ever generated. Only the ensemble statistics of δX_j are available which means only a statistical or effective velocity correction "$E \left[\Delta V_{N_j} \right]$" can be computed. In the STEAP error analysis program, this effective velocity correction is assumed to have form:

$$\text{"}E \left[V_{N_j} \right]\text{"} = \rho_j \; \frac{\alpha j}{|\alpha j|} .$$

The magnitude ρ_j is given by the Lee-Boain analytic solution as described in the analysis of subroutine DUSTAT.

The direction of the effective velocity correction is assumed to coincide with the eigenvector corresponding to the maximum eigenvalue of S_j. This eigenvector is denoted by α_j.

After the updated control covariance $P^+_{c_j}$ has been computed, the target condition covariance matrix W^+_j following the guidance correction is computed using the equation:

$$W^+_j = \eta_j \, P^+_{c_j} \, \eta^T_j$$

where variation matrix η_j has been previously computed in subroutine GUID.

2. Final Insertion

A final insertion event describes an insertion (into halo orbit) which may be accomplished by an impuslive or by a finite burn. If the burn is impulsive, the expected correction has been input as REXV. This vector is used to compute the executive error matrix, $\tilde{Q}_F$, just as ΔV_{N_j} is used to compute $\tilde{Q}_j$.

If the burn is finite, GUIDM calls NTM and PSIM to compute the state to state transition matrix, $\Phi(t_F, t_B)$, and the control to final state transition matrix $\theta\ (t_F, t_B)$, where the control parameters are the pointing angles α and β and the thrust magnitude T. Both the control and the knowledge covariances are propagated by $\Phi(t_P, t_B)$

$$P^-_{c_F} = \Phi(t_F, t_B) \, P^+_{c_B} \, \Phi(t_F, t_B)^T$$

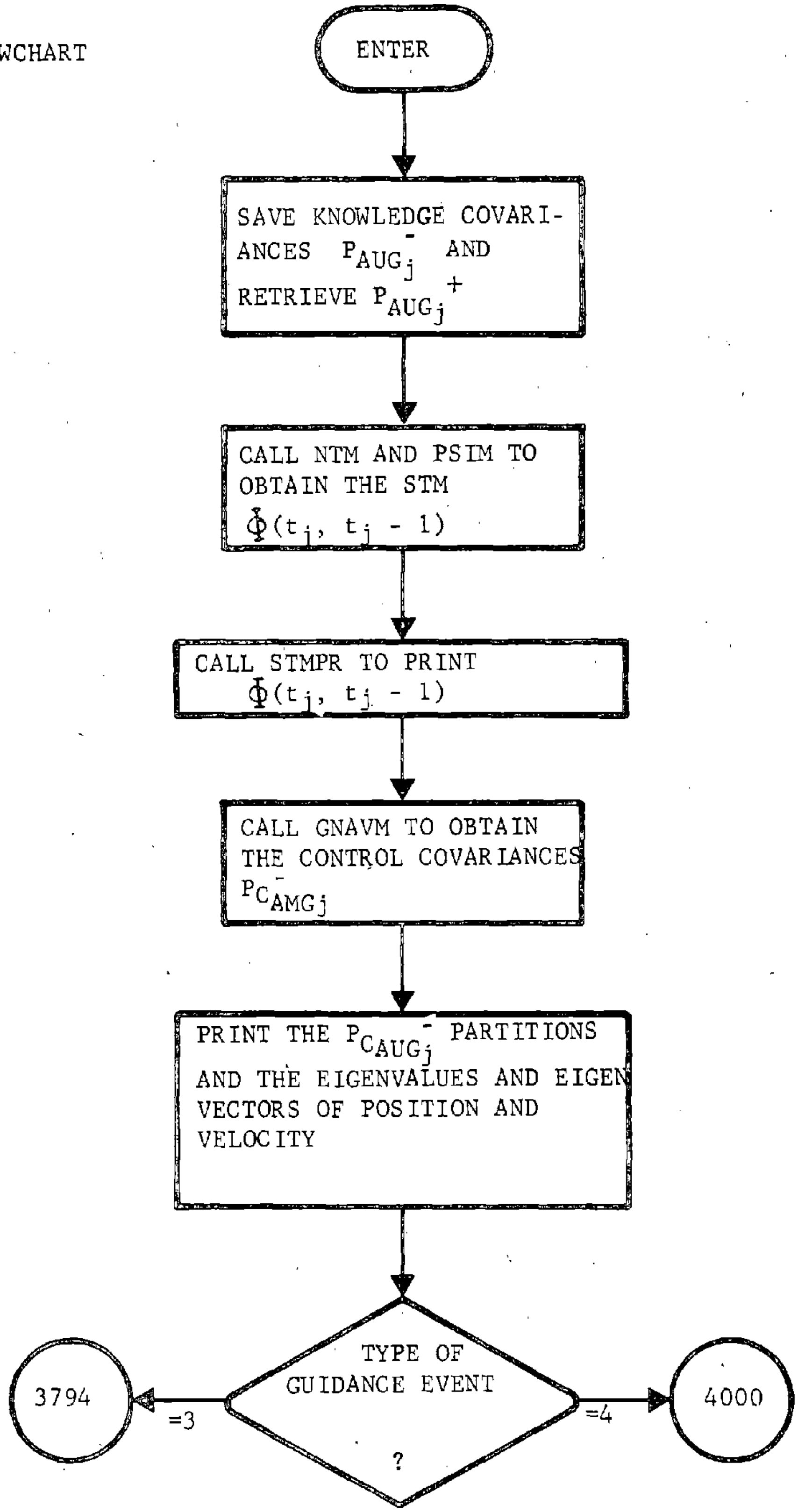
$$P^-_{K_F} = \Phi(t_F, t_B) \, P^+_{K_B} \, \Phi(t_F, t_B)^T$$

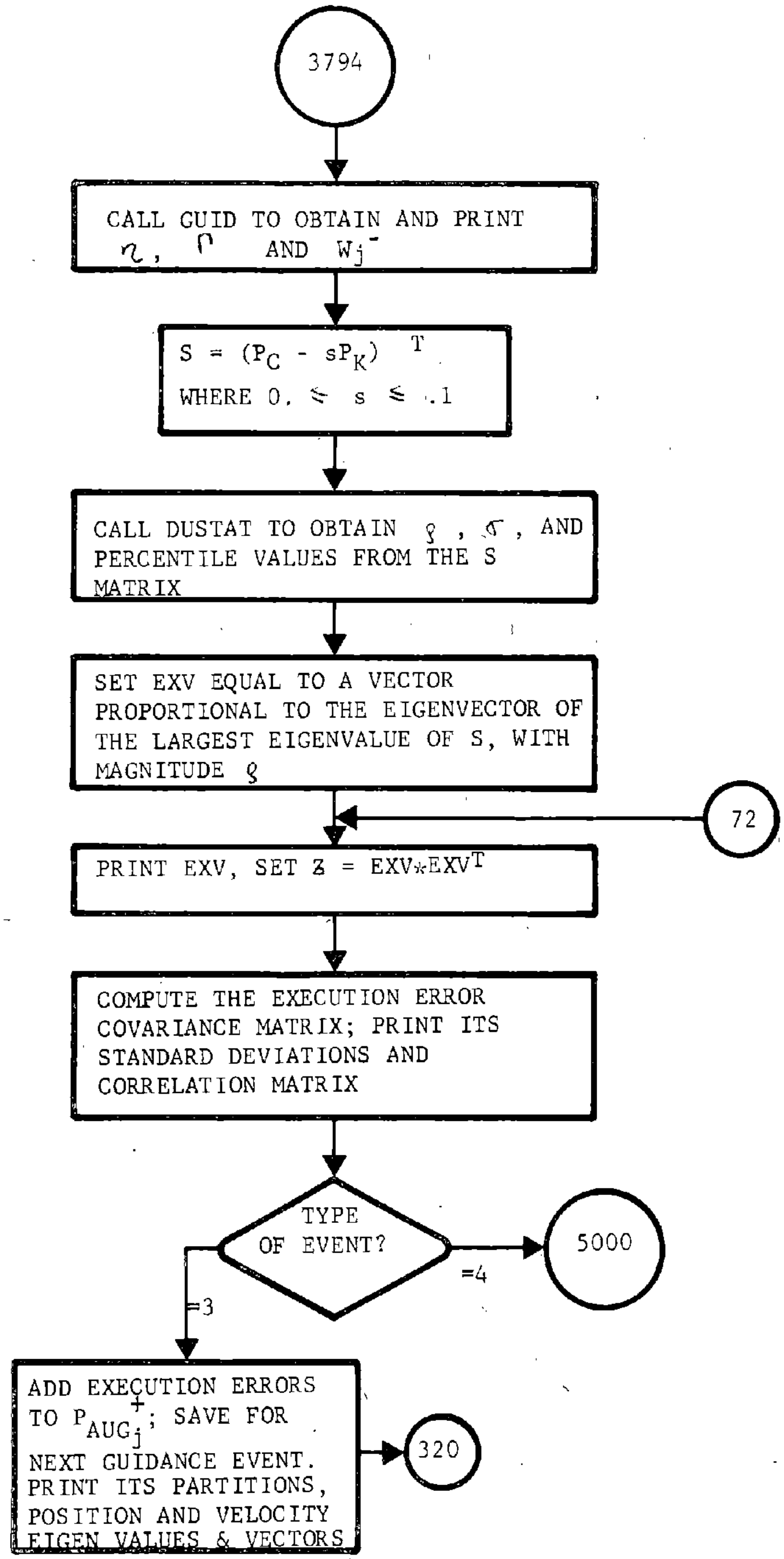
The execution error matrix for the finite burn is the 6 x 6 matrix

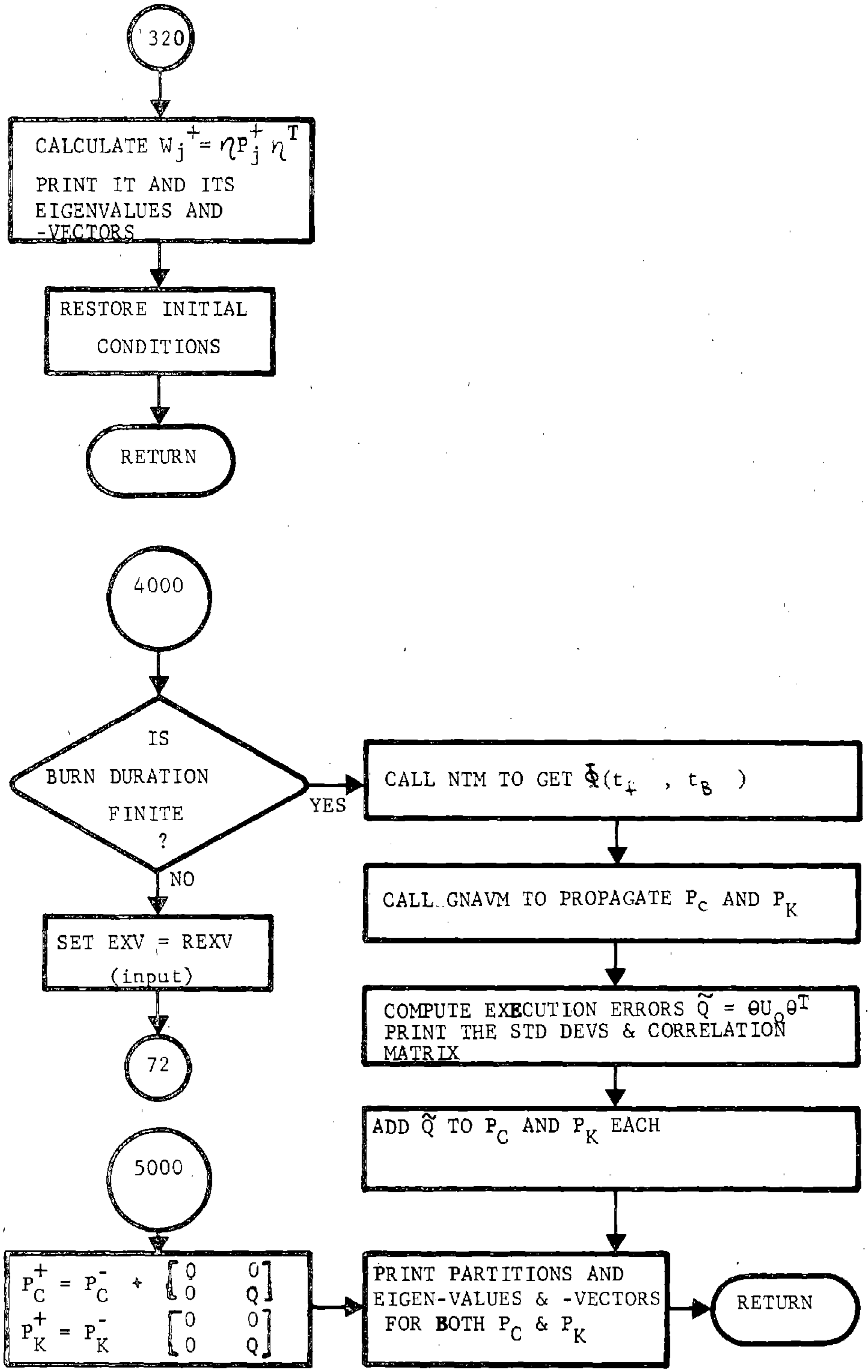
$$\tilde{Q}_F = \theta\ (t_F, t_B) \, U_o \, \theta\ (t_P, t_B)^T$$

where U_o is the diagonal matrix whose diagonal elements are $\sigma_\alpha{}^2$, $\sigma_\beta{}^2$, and $\sigma_T{}^2$; U_o is input to the program. In either, the impulsive or the finite case, both the knowledge and the control covariances are updated by adding the execution errors. These results are printed, and the end of the final insertion event is the end of the ERRAN run.

GUIDM FLOWCHART




3794
CALL GUID TO OBTAIN AND PRINT η, Γ AND W_j^-
$S = (P_C - sP_K)^T$
WHERE $0. \leq s \leq .1$
CALL DUSTAT TO OBTAIN ρ, σ, AND PERCENTILE VALUES FROM THE S MATRIX
SET EXV EQUAL TO A VECTOR PROPORTIONAL TO THE EIGENVECTOR OF THE LARGEST EIGENVALUE OF S, WITH MAGNITUDE ρ
72
PRINT EXV, SET $Z = EXV*EXV^T$
COMPUTE THE EXECUTION ERROR COVARIANCE MATRIX; PRINT ITS STANDARD DEVIATIONS AND CORRELATION MATRIX
TYPE OF EVENT?
=4
5000
=3
ADD EXECUTION ERRORS TO $P_{AUG_j}^+$; SAVE FOR NEXT GUIDANCE EVENT. PRINT ITS PARTITIONS, POSITION AND VELOCITY EIGEN VALUES & VECTORS
320


320
CALCULATE $W_j^+ = \eta P_j^+ \eta^T$
PRINT IT AND ITS EIGENVALUES AND -VECTORS
RESTORE INITIAL CONDITIONS
RETURN
4000
IS BURN DURATION FINITE ?
YES
NO
CALL NTM TO GET $\Phi(t_f, t_B)$
CALL GNAVM TO PROPAGATE P_C AND P_K
COMPUTE EXECUTION ERRORS $\tilde{Q} = \theta U_Q \theta^T$
PRINT THE STD DEVS & CORRELATION MATRIX
ADD $\tilde{Q}$ TO P_C AND P_K EACH
SET EXV = REXV (input)
72
5000
$P_C^+ = P_C^- + \begin{bmatrix} 0 & 0 \\ 0 & Q \end{bmatrix}$
$P_K^+ = P_K^- \begin{bmatrix} 0 & 0 \\ 0 & Q \end{bmatrix}$
PRINT PARTITIONS AND EIGEN-VALUES & -VECTORS FOR BOTH P_C & P_K
RETURN

SUBROUTINE HGIDNS

PURPOSE: TO COMPUTE THE CHANGE REQUIRED TO THE CONTROL VARIABLES FOR TARGETING

CALLING SEQUENCE: CALL HGIDNS

ARGUMENTS:
NONE

LOCAL SYMBOLS:

TFP	TIME FROM PERIAPSIS (SECS)
E1	SEMI-MAJOR AXIS (KM)
E2	ECCENTRICITY
E4	ARGUMENT OF PERIAPSIS (DEG)
DF	COMPUTED INJECTION TIME (DAYS)
AETA	SENSITIVITY PARTIALS OF TARGETS WRT POSITION CHANGES
BETA	SENSITIVITY PARTIALS OF TARGETS WRT VELOCITY CHANGES
ETAI	TARGETING MATRIX GAMMA
STMB	PARTION OF STATE TRANSITION MATRIX OF POSITION CHANGES WRT VELOCITY CHANGES
STMD	PARTION OF STATE TRANSITION MATRIX OF VELOCITY CHANGES WRT VELOCITY CHANGES
TEMX1	SENSITIVITY MATRIX
PXU	PARTIALS OF STATE WRT FINITE BURN CONTROLS AT THE END OF THE BURN
TXU	PARTIALS OF STATE WRT FINITE BURN CONTROLS AT TARGET TIME
ATAR	ACTUAL TARGET VECTOR ON CURRENT NOMINAL TRAJECTORY
AER	TARGET ERROR VECTOR (DESIRED-ACTUAL)
DELTAV	CONTROL UPDATE VECTOR
XFEQ	INJECTION STATE VECTOR IN EARTH EQUATORIAL-GEOCENTRIC COORDINATES
TCHANG	ACCUMULATED TOTAL CHANGE IN CONTROL VECTOR

COMMON USED:

DL	STM	DTOL	SPD
XB	ACCTH	PERT	SMU(4)
TB	IBURN	LIBR	H
XF	ITMAX	PI	ECEQ
ASTM	DTAR	RPD	

COMMON COMPUTED:

XL	ALPHA0
ITOL	BETA0
ITER	TBURN
KWIT	

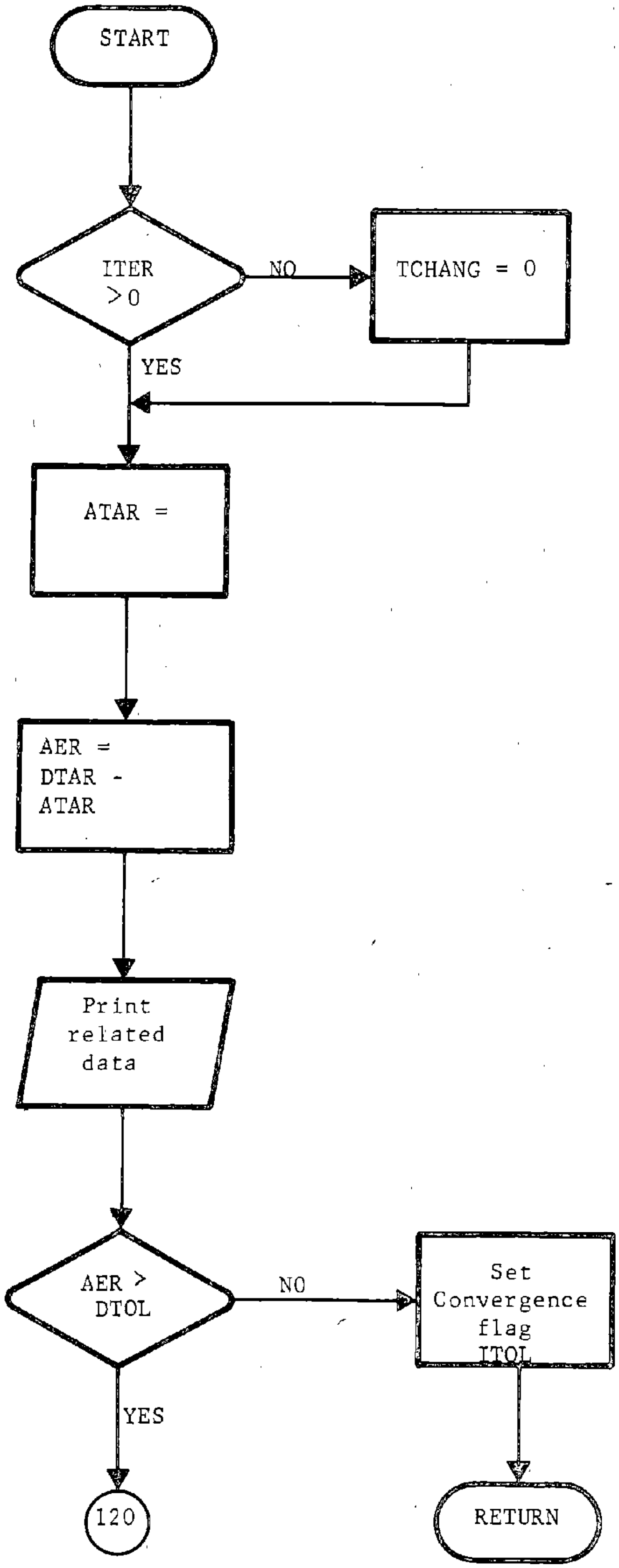
HGIDNS FLOW CHART

Zero out accumulated change first time through

Compute actual target from nominal trajectory

Compute target error

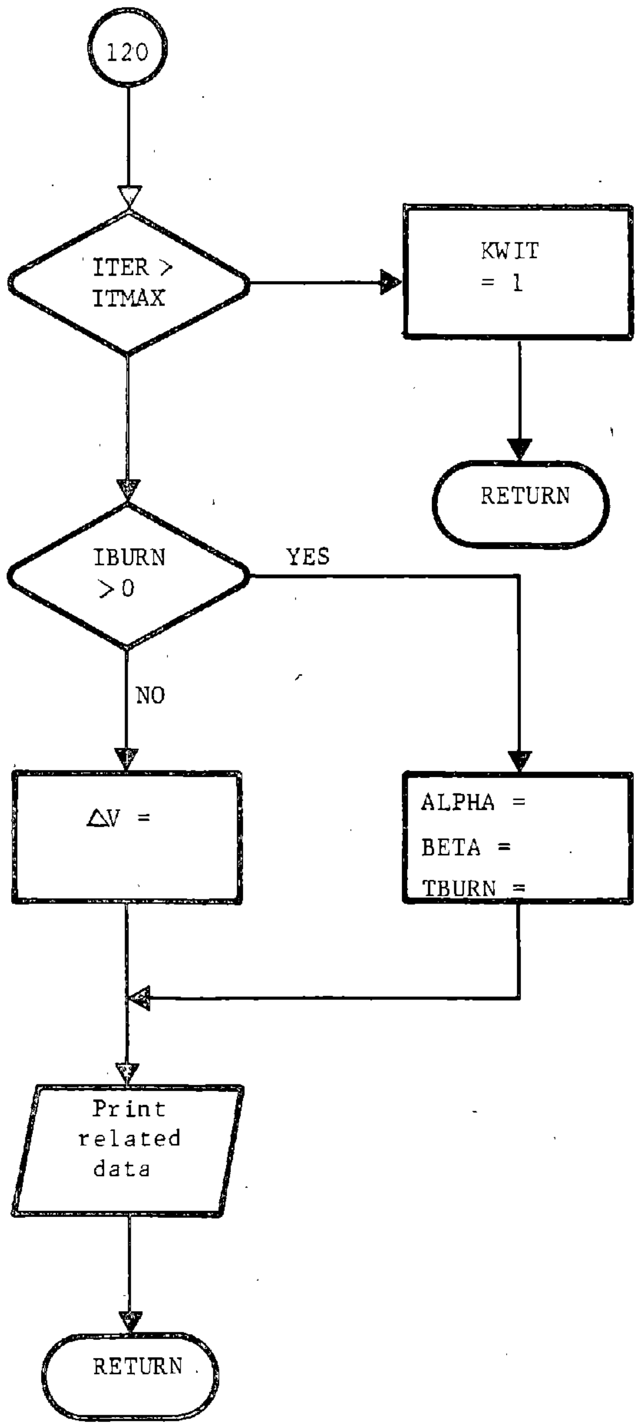
Error greater than tolerance



Terminate case if maximum iterations are exceeded

Is this a finite burn case

Generate update control variables



SUBROUTINE HLAUCH

PURPOSE: TO COMPUTE THE INJECTION TIME

CALLING SEQUENCE: CALL HLAUNCH(X,DJ)

ARGUMENTS:

X	I	INJECTION STATE VECTOR IN ECLIPTIC-GEOCENTRIC COORDINATES (X, Y, Z (KM) AND XDOT, YDOT, ZDOT (KM/SEC))
DJ	I/O	INPUT AS DESIRED INJECTION JULIAN DATE, OUTPUT AS ACTUAL INJECTION JULIAN DATE

LOCAL SYMBOLS:

REFJD	JULIAN DATE JAN 1, 1950.0
DLA	EARTH EQUATORIAL DECLINATION OF ANGULAR MOMENTUM VECTOR (RAD)
SDLA	SIN OF 'DLA'
PHILSR	LATITUDE OF LAUNCH SITE (RAD)
SIGR	LAUNCH AZIMUTH RADIANS
SSIG	SINE OF 'SIGR'
SPHI	SINE OF 'PHILSR'
CPHI	COSINE OF 'PHILSR'
CSIG	COSINE O 'SIGR'
TD	DAYS FROM REFJD TO INJECTION DAY
GHA	GREENWICH HOUR ANGLE (DEG)
CTHE	COSINE OF 'THE'
STHE	SINE OF 'THE'
THE	RIGHT ASCENSION AT LAUNCH (RAD)
TL	LAUNCH TIME ON DAY OF LAUNCH (DAYS)
FL	TRUE ANOMALY OF LAUNCH SITE (RAD)
PSIB	ANGLE BETWEEN LAUNCH AND INJECTION (RAD)
TC	COAST TIME (SEC)
TB	TIME BETWEEN LAUNCH AND INJECTION (DAYS)
SIG	LAUNCH AZIMUTH (DEG)
TI	INJECTION TIME OF DAY (DAYS)
XEQ	INJECTION STATE VECTOR IN EQUATORIAL-GEOCENTRIC COORDINATES (X, Y, Z (KM) AND XDOT, YDOT, ZDOT (KM/SEC))
ELEMS	6 VECTOR OF ORBITAL ELEMENTS (SEMI-MAJOR AXIS (KM), ECCENTRICITY, INCLINATION(DEG), LONGITUDE OF ASCENDING NODE(DEG), ARGUMENT OF PERIAPSIS (DEG), TRUE ANOMALY(DEG))
WHAT	3 ELEMENT ANGULAR MOMENTUM VECTOR IN EQUATORIAL-GEOCENTRIC COORDINATES
ZAXIS	PSEUDO POLE VECTOR (0., 0., 1.)

SUBROUTINES REQUIRED:

DMATPY	CALJUL
DUXV	DUNIT
DVECRD	DANGV2
CAREL	

SUBROUTINE HLAUCH (CONTINUED)

COMMON USED:

FI	THELS	ECEQ
PSI1	PHILS	PI
PSI2	THEDOT	TWOPI
TIM1	RPRAT	RPD
TIM2	SIGMAL	SPD

HLAUCH Analysis

HLAUNCH computes the injection time from the injection state and the launch profile parameters input by the user.

The injection state is first rotated from the ecliptic coordinate system, which was input, to the earth equatorial system.

$$\underline{R}cA = \Phi Eceq \quad \underline{R}ec$$
$$\underline{V}cA = \Phi Eceq \quad \underline{V}ec \tag{1}$$

The unit normal to the launch/orbit plane is then calculated in earth equatorial coordinates as

$$W_T = \frac{R_{CA} \times V_{CA}}{|R_{CA} \times V_{CA}|} \tag{2}$$

The inclination of the orbit plane i (= arc cos Wz) should equal the desired input value. The orbit plane inclination must equal or exceed the latitude of the launch site $_L$ to permit a coplanar parking orbit and transfer orbit as indicated in Figure 19. In the case that sin i sin $_L$ the launch azimuth is defined by

$$\sin \Sigma_L = \frac{\cos i}{\cos \Phi_L} \tag{3}$$

and the solution with $0 \leq \Sigma_L \leq 90$ degress is selected. In this case the parking orbit nominal is identical to that of the transfer plane given by (2).

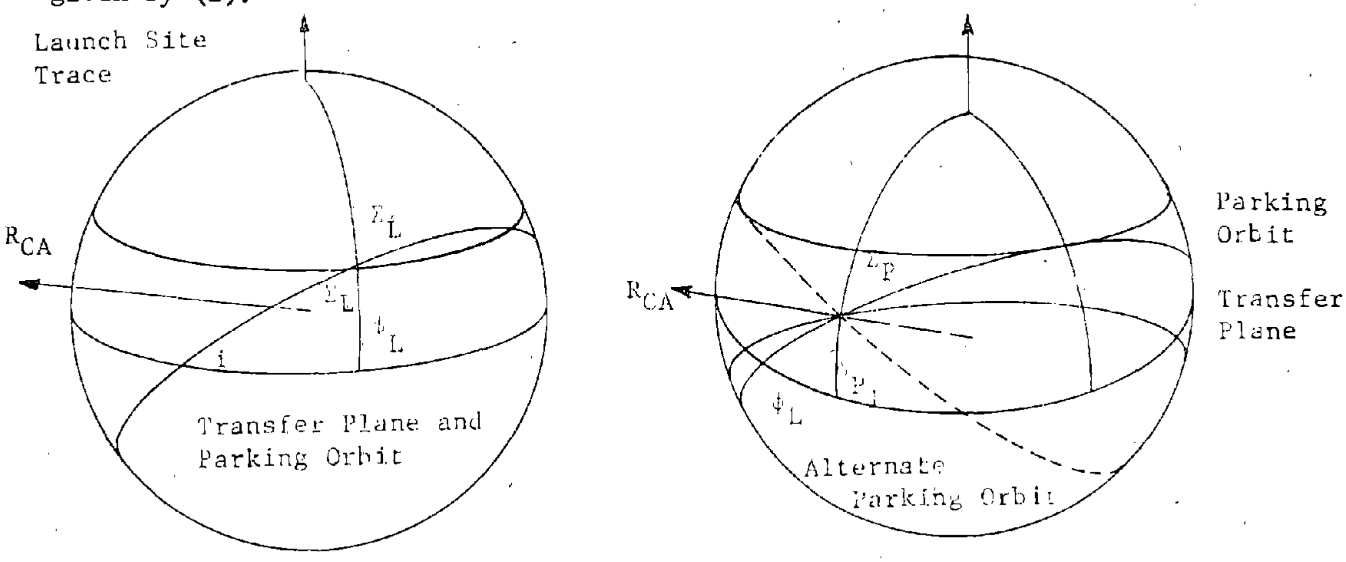


Figure 1 Transfer Plane/Parking Orbit Geometry

If $|\sin i| < |\sin \phi_L|$, the parking orbit and the transfer orbit cannot be coplanar (Figure 16). In this case the parking orbit is defined to be in the plane have a launch azimuth of Σ_L = 90 deg., containing the closest approach radius vector R_{CA}, and nearest the transfer plane. (Note the alternate parking orbit plane in Figure 1b which also satisfies the first two of these requirements). The unit normal to the parking orbit plane is given by

$$W_p = \frac{R_{CA} \times V_p}{|R_{CA} \times V_p|} \tag{4}$$

where V_p is the velocity vector at the injection point in the parking orbit. V_p is given by

$$V_p\ (\Sigma_p) = \begin{array}{l} -\cos\ \theta_p\ \sin\ \delta_p\ \cos\ \Sigma_p\ -\sin\ \theta_p\ \sin\ \Sigma_p \\ -\sin\ \theta_p\ \sin\ \delta_p\ \cos\ \Sigma_p\ +\cos\ \theta_p\ \sin\ \Sigma_p \\ \cos\ \delta_p\ \cos \Sigma \end{array}$$

where (θ_p, δ_p) are the equatorial right ascension and declination of the periapsis position R_{CA}. For the specific parking orbit plane having Σ_L = 90 deg, including R_{CA}, and nearest the transfer plane $_p$ must satisfy

$$\sin\ _p = \frac{\cos\ \phi_L}{\cos\ \delta_p} \tag{6}$$

$$\text{sgn}\ (\cos\ \Sigma_p) = \text{sgn}\ \ V_{CA}\ \cdot\ V_p(0)$$

where $0 \leq \Sigma_p \leq 180$ deg and where the equation (5) is used.

Thus the unit normal to the parking orbit plane may be computed by either (2) or (4) and the launch azimuth is either given by (2.47) or Σ_L = 90 deg. In either case the remaining calculations proceed as follows. the right ascession at launch $Ⓗ_L$ is defined by.

$$\cos \textcircled{H}_L = \frac{W_x \sin \Phi_L \sin \Sigma_L + W_y \cos \Sigma_L}{W_z^2 - 1} \tag{7}$$

$$\sin \textcircled{H}_L = \frac{W_y \sin \Phi_L \sin \Sigma_L - W_x \cos \Sigma_L}{W_z^2 - 1}$$

The launch date input by the user is recalculated as the integer day (0^h ut) closest to the inital date input by the user. The Greenwich hour angle at 0^h ut of the launch date is then

$$GHA = 100^{\circ}.07554260 + 0^{\circ}.9856473460 \quad T_d \qquad 2^{\circ}.9015 \times 10^{-13} \; T_d^2 \tag{8}$$

The launch time on the day of launch is

$$t_L = \frac{\textcircled{H}_L - \theta t - GHA) \quad \text{mod } 2\pi}{w}$$

where w is the rotation rate of the launch planet and $_L$ is the longitude of the launch site, both being read in as input.

The unit vector toward the launch position is the

$$R_L = (\cos \Phi_1 \cos \textcircled{H}_L|, \cos \Phi_L \sin \textcircled{H}_L|, \sin \Phi_L|) \tag{10}$$

The true anomaly of the launch site f_L is calculated as:

$$\cos f_L = R_L \cdot R_{CA}$$

$$\sin f_L = R_L \cdot V_{CA} \tag{11}$$

The angle between launch and injection is

$$\psi_B = 2\pi - f_L \tag{12}$$

The coast time t_C may now be computed

$$t_C = \left[\psi_B - \left(\psi_1 + \psi_2\right)\right] \; k\Phi \tag{13}$$

where $_1$ and $_2$ are the angle of $^{\circ}$ the first and second burns and $k\Phi$ is the inverse parking orbit coast rate, all of which are input.

The time between launch and injection is therefore

$$t_B = t_1 + t_2 + t_C \tag{14}$$

where t_1 and t_2 are the input time durations of the first and second burn

The injection time is then

$$t_I = t_L + t_B \tag{15}$$

SUBROUTINE HPRELM

PURPOSE: TO INITIALIZE CONSTANT SAND DEFAULT VALUES, READ INPUT DATA. AND CALCULATE THE ZERO ITERATE GUESS.

CALLING SEQUENCE: CALL HPRELM

ARGUMENTS:
NONE

LOCAL SYMBOLS:

BTIME	DURAION OF FINITE BURN (DAYS)
IBIAS	INPUT FLAG INDICATING THAT A BIAS VECTOR IS TO BE ADDED TO THE LIBRATION STATE PRIOR TO TARGETING
IZERO	INPUT FLAG INDICATION THE SOURCE OF THE INITIAL CONDITIONS
XINT	INJECTION STATE VECTOR TO BE INTEGRATE IF ITMAX=0
DTAR	ARRAY OF TARGET VALUES
ZDAT	INITIAL GUESS OF SPACECRAFT VELOCITY AT LIBRATION POINT, IF IZERO=5
RE	RADIUS VECTOR FROM THE SUN TO THE EARTH

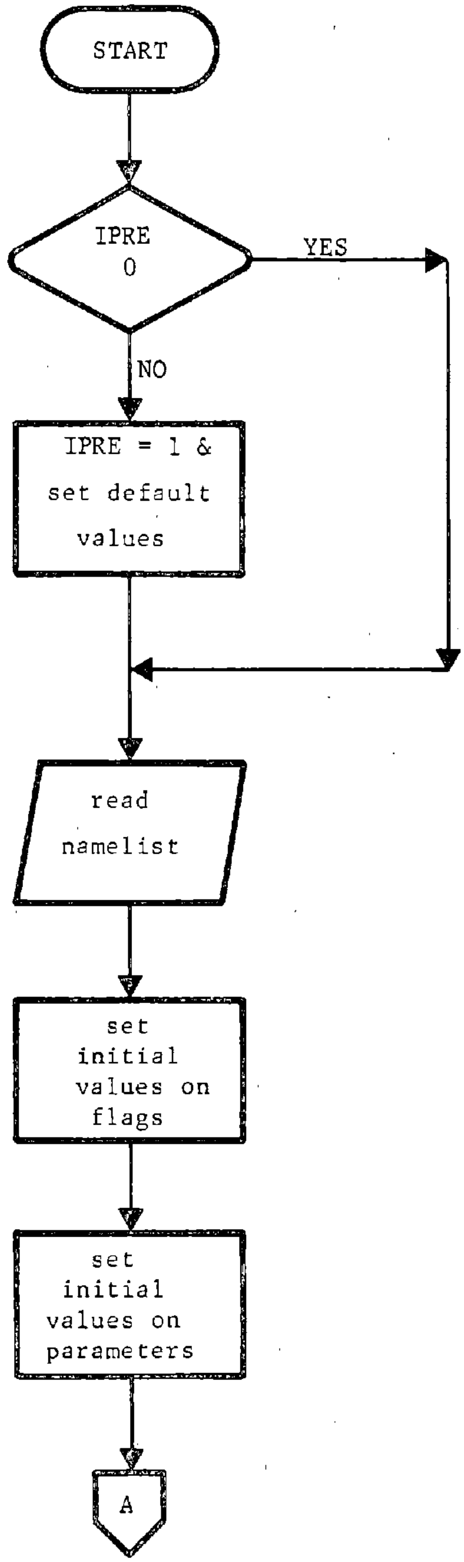
SUBROUTINES REQUIRED:

DZERO	TRNSPS
CALJUL	DSHIFT
PECEQ	EPHGT
DVSMLT	DAVECT
DVECRD	HZERIT

COMMON COMPUTED:

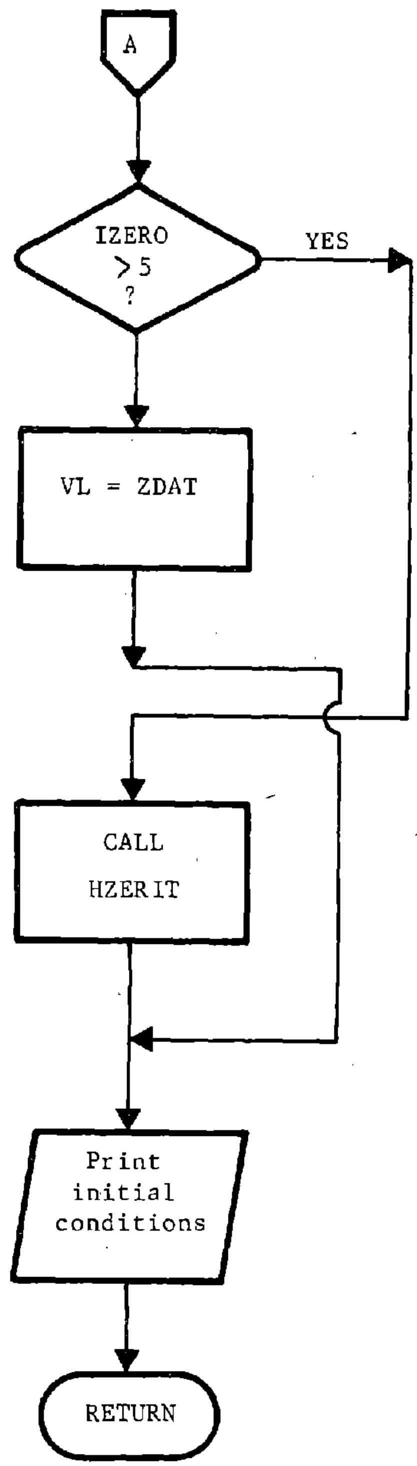
ALPHA	FI	ITMAX	NB	PERT
ATRY	IBTYPE	LAUNCH	NBOD	PHILS
BETA	IDISK	LIBR	NFR	PSI1
DTOL	IPRE	MSGLVL	NPOINT	PSI2
RLIBR	TDUR	TIM1	XISP	ITOL
RPRAT	THEDOT	TIM2	ZBIAS	KWIT
SCMASS	THELS	TMPR	IDON	RETRO
SIGMAL	THRMAG	TP	ITER	LNON
IBURN	ECEQ			
TBURN	XF			
TB	VOLP			
DL	XL			

HPRELM FLOWCHART



Initial velocity of S/C at libration point comes from input

Initial velocity comes from tables or Lamberts solution



SUBROUTINE HTRJTY

PURPOSE: TO CONTROL THE TRAJECTORY GENERATION PHASE

CALLING SEQUENCE: CALL HTRJTY

ARGUMENTS:
NONE

LOCAL SYMBOLS:

DJ	JULAIN DATE OF INJECTION
TBIAS	BIAS DATE ADDED TO PRINT OUT TIME
TCOAST	COAST TIME
SH	SIGN OF INTEGRATION STEP
TTO	PRINT TIME (DAYS)
TACT	PRINT TIME OF NEXT SPECIAL PRINT POINT (DAYS)
TTOPRE	PRINT TIME OF PREVIOUS PRINT POINT (DAYS)
ST	SIGN OF PRINT POINT
TTP	DAYS FROM INJECTION
DJP	JULIAN DATE OF PRINT POINT
RMAG	MAGNITUDE OF RADIUS VECTOR (KM)
VMAG	MAGNITUDE OF VELOCITY VECTOR (KM/SEC)
RA	RIGHT ASCENSION (DEG)
DEC	DECLINATION (DEG)
DELV	MAGNITUDE OF IMPULSIVE BURN DELTA V VECTOR
KSWB	FLAG TO INDICATE INITIATION OF FINITE BURN
IRONG	NUMBER OF ERRORS ENCOUNTERED DURING TRAJECTOR PRINT PHASE
KPOINT	CURRENT SPECIAL PRINT POINT NUMBER
ISTOP	FLAG TO INDICATE ARRIVAL OF STOPPING CONDITIONS
KSWP	FLAG TO INDICATE THAT CURRENT PRINT POINT IS A SPECIAL PRINT POINT
X	STATE VECTOR

SUBROUTINES REQUIRED:

HLAUCH	EPHGT	DVSMLT	DAVECT	SET1
PSTART	MSTART	ORBINT	COWELL	DSHIFT
CALJUL	EVAL	DSVECT	DABSV	DRD
DMATPY	ORBEND	BURN		

COMMON USED:

NBOD	TMPR	THRMAG
NB	PI	XISP
IARRAY	RPD	SCMASS
ITOL	SPD	
ITER	H	
CDTAR	LAUNCH	
IDISK	ECEQ	
NPOINT	MSGLVL	
TP	PLANET	

SUBROUTINE HTRJTY (CONTINUED)

COMMON COMPUTED/USED:

XL	VOLP
DL	IBURN
XB	IDON
TB	LNON
XF	
TOUR	

COMMON COMPUTED:

BSTM	KWIT
ASTM	ALPHA
STM	BETA
ACCTH	TBURN

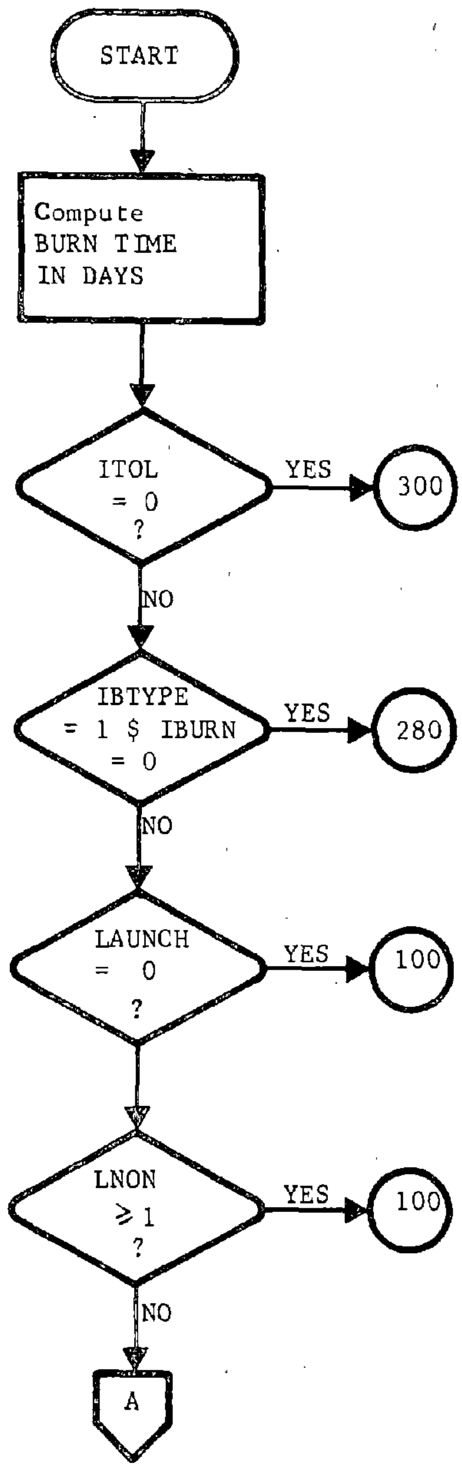
HTRJTRY FLOWCHART

Check if targeting has converged

Check if finite burn is being targeted from impulsive burn

Should run target based on actual launch profile

Check if actual launch profile has been generated

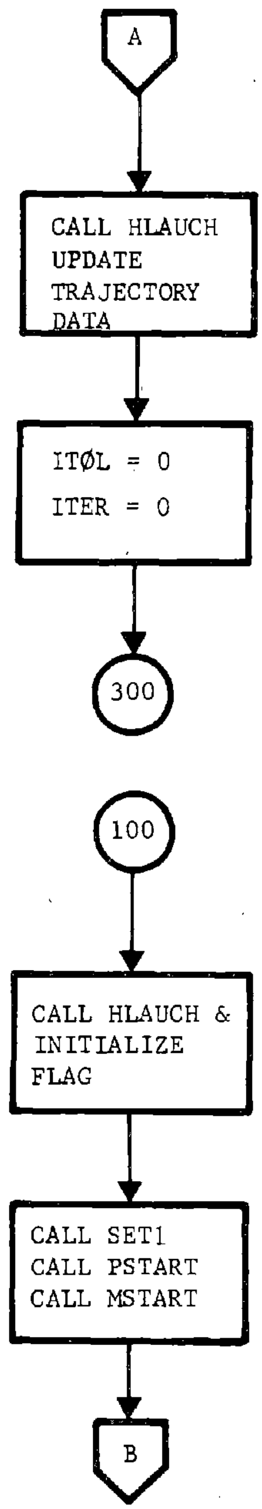


Generate actual launch profile

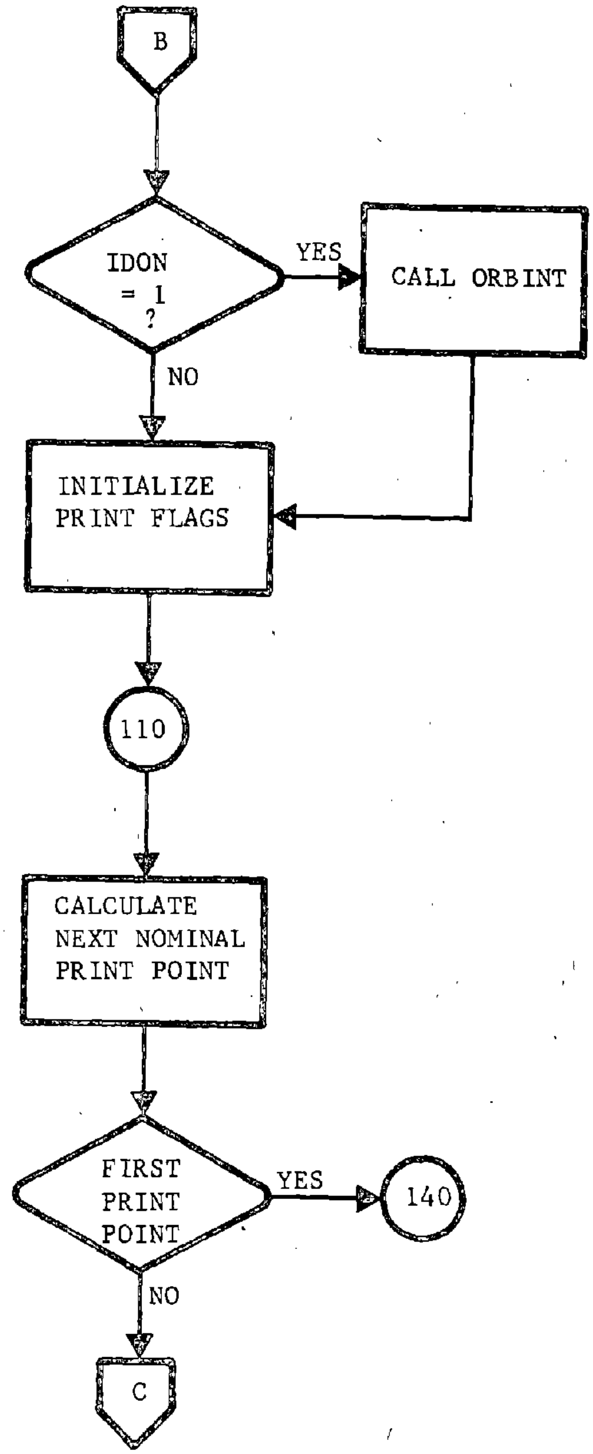
Reset convergence flag & iteration counter

Generate projected launch profile

Initialize integrator for COAST phase of trajectory



Check if orbit file is being written



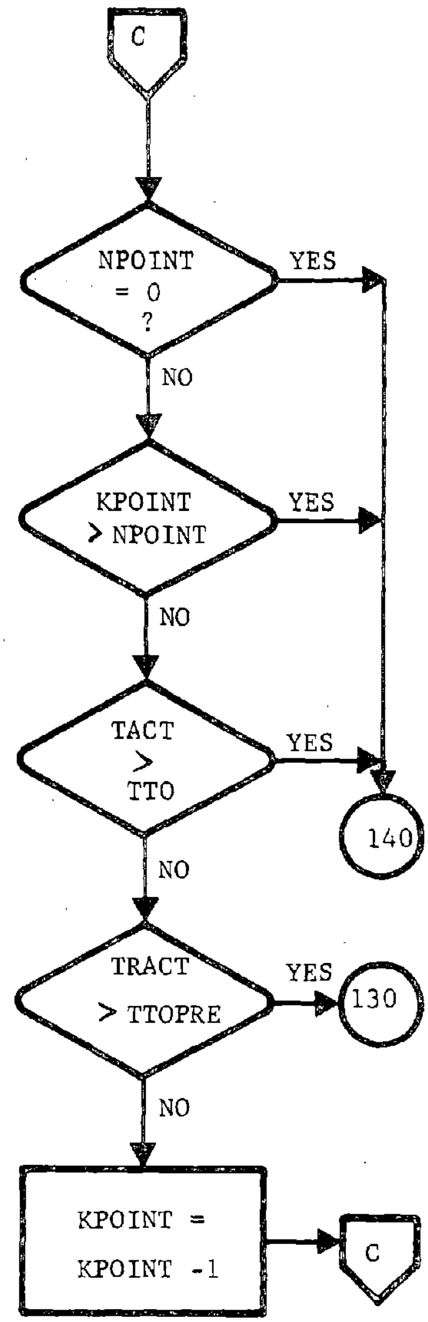
Skip special print logic if first point

Check if there are special print points

Has last special print been used

Is special print beyond nominal print point

Is special point beyond last point



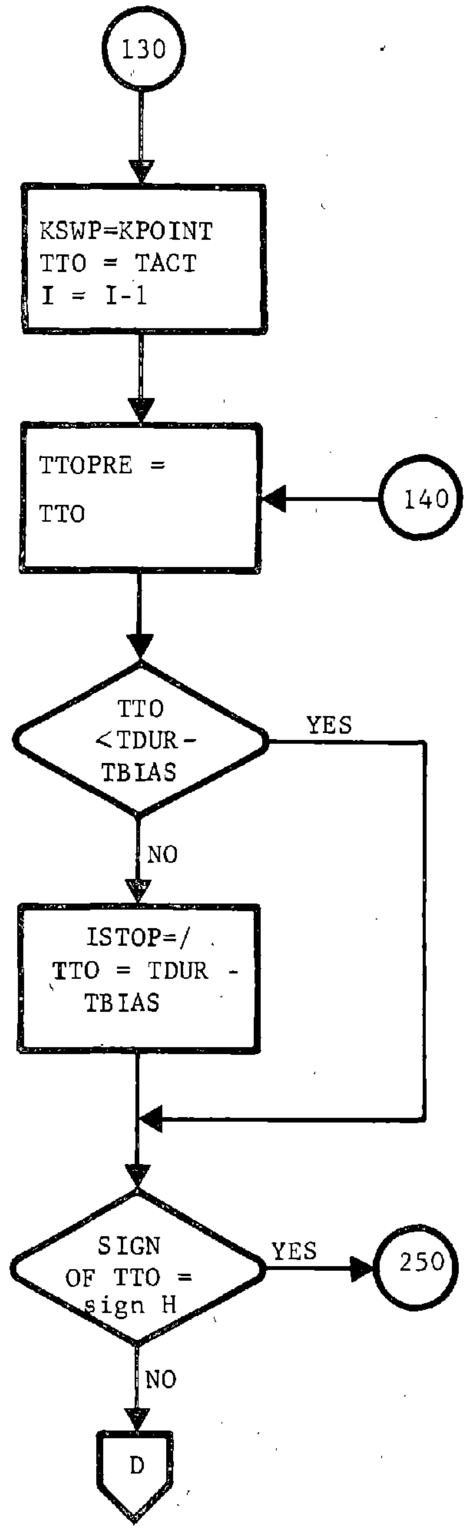
Set special print flag, set current print point to special print point, decrement index for NOMNAL point

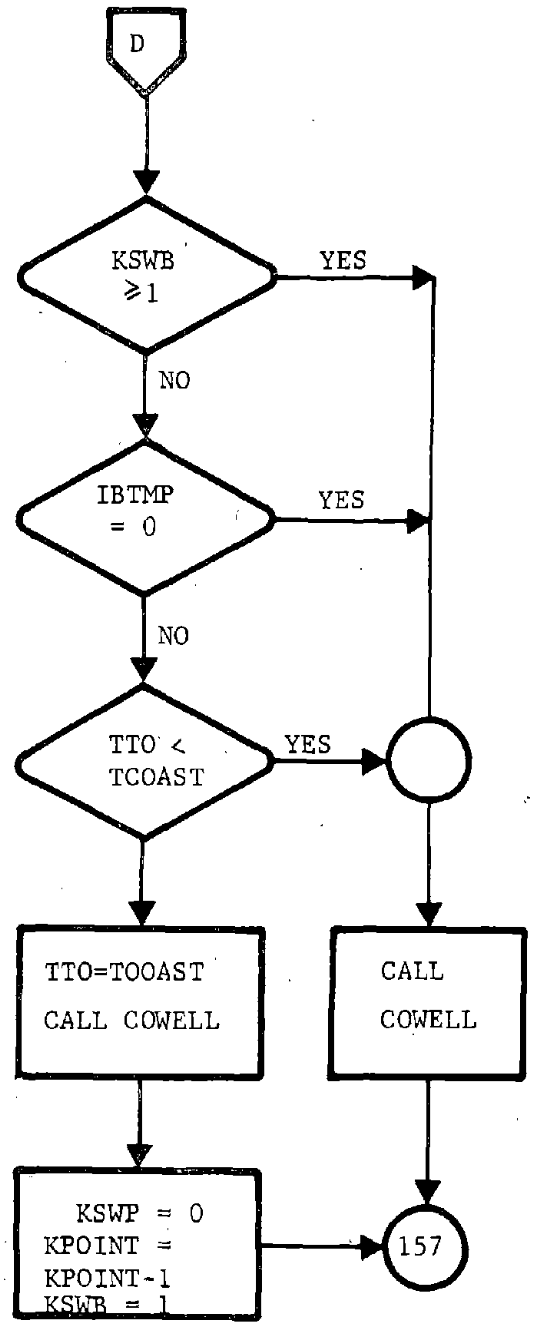
Set previous print point to current print point

Check to see if current print point is within section

Reset print point to terminal value & set termination flag

Check if current print point is outside integration range





Are we in the burn phase

Is this an impulsive burn case

Is current print point within coast phase

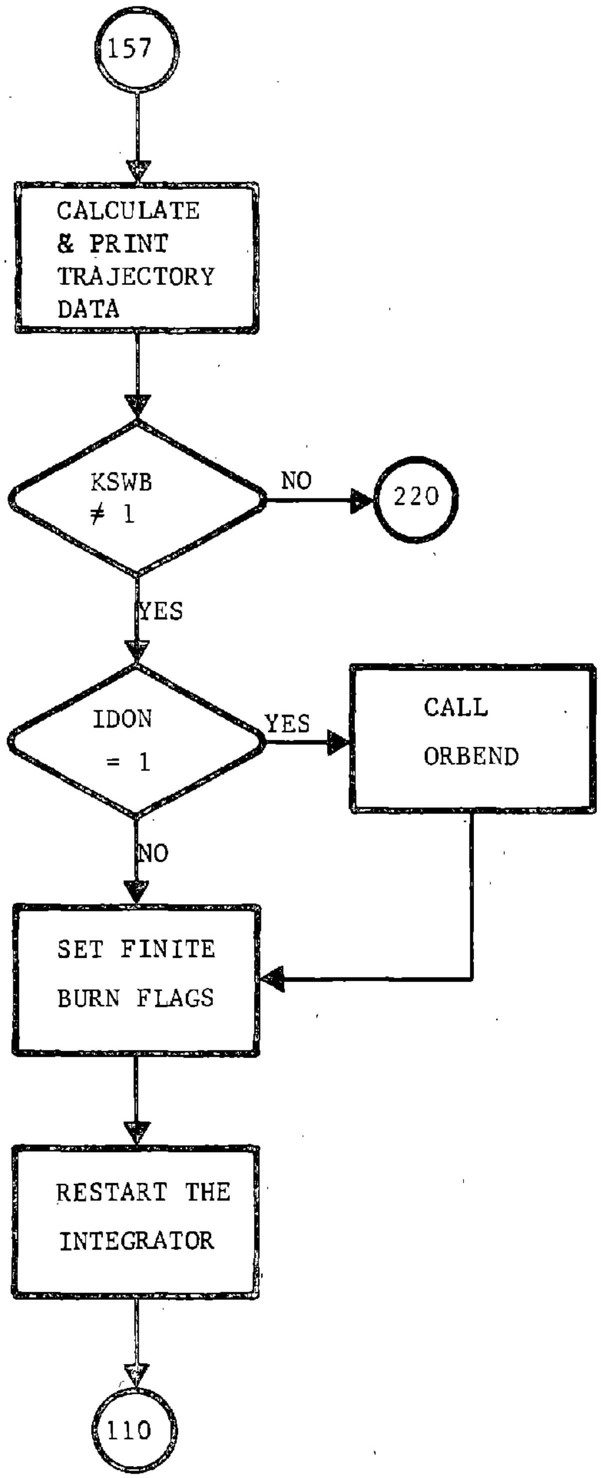
Integrate to burn initiation

Integrate to print point

Reset flags

Must finite burns be initiated

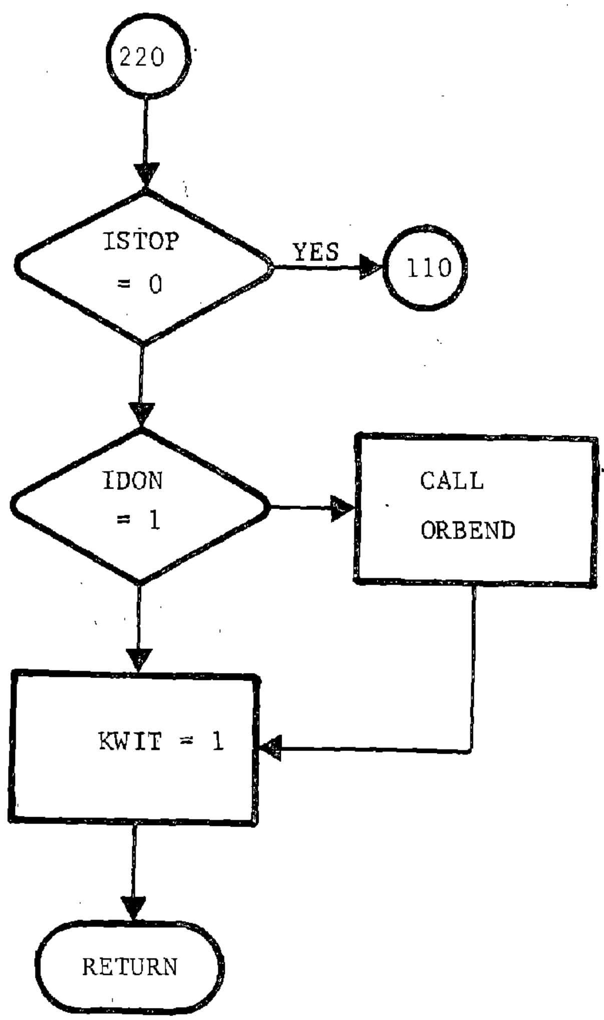
Write remaining acceleration vectors if necessary



Is there more data to be generated

Write remaining acceleration vectors if necessary

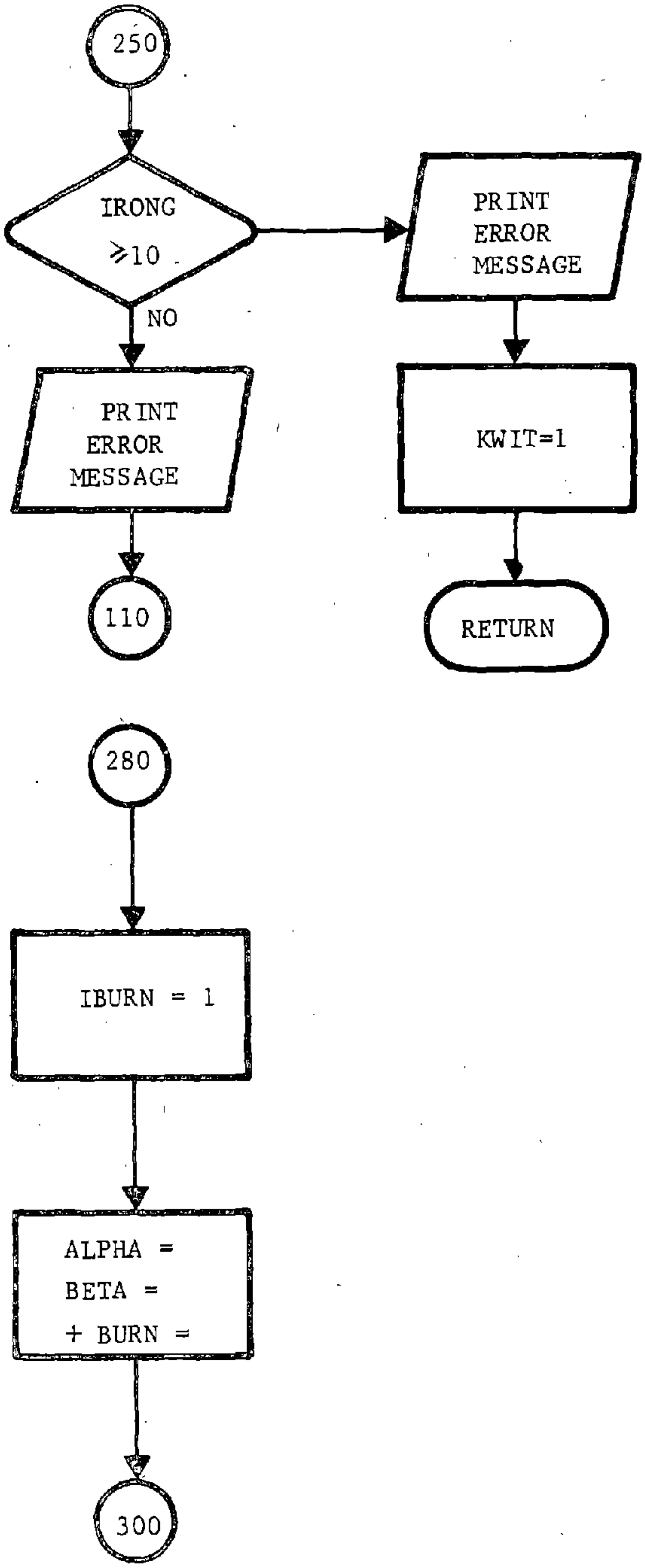
Set case termination flag



Have 10 bad print point accumulated

Set burn flag

Determine finite burn parameters from impulsive burn data



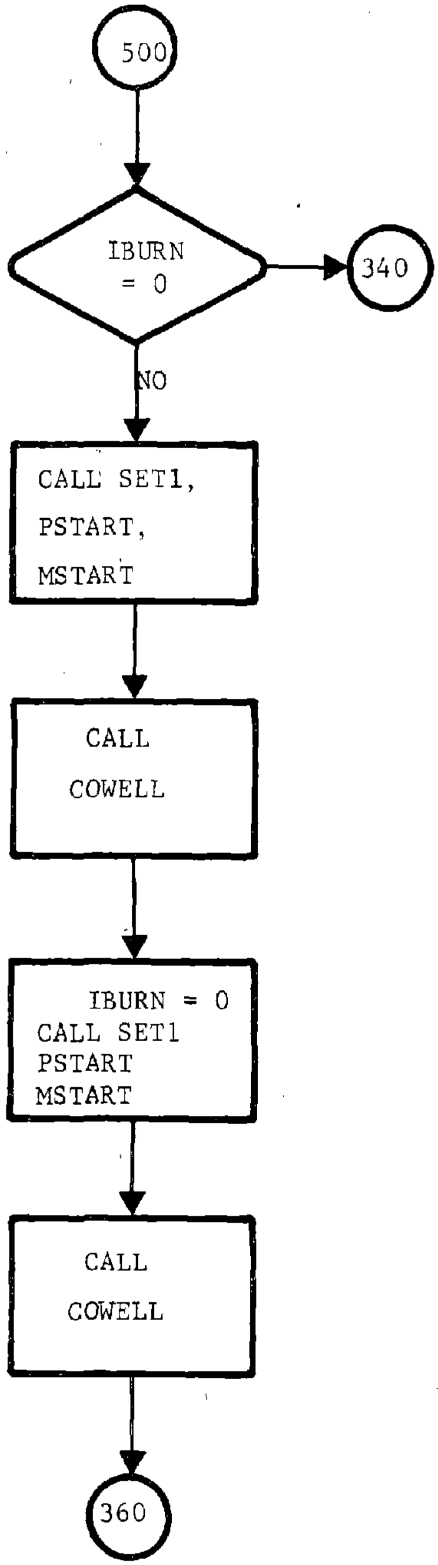
Is this an impulsive burn

Initialize integrator

Integrate burn phase of trajectory

Turn burn flag off & restart integrator

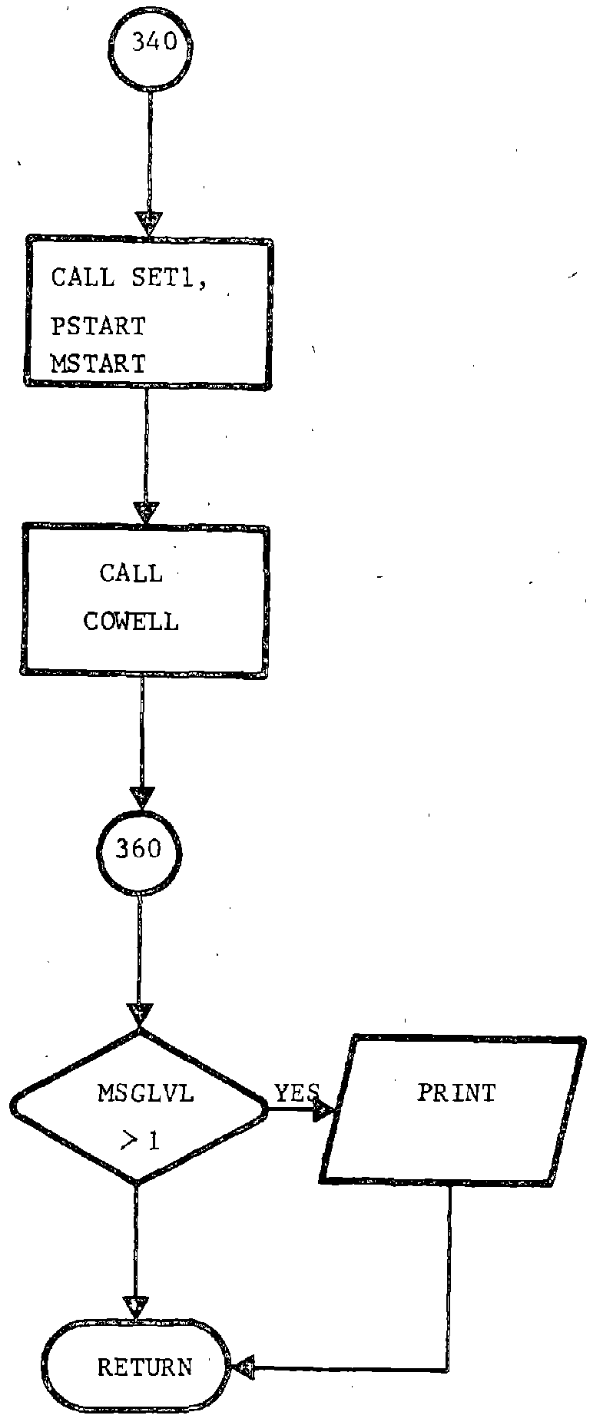
Integrate coast phase of trajectory



Initialize integrator

Integrate trajectory for impulsive burn case

Print debug data if necessary



SUBROUTINE HZERIT

PURPOSE: TO COMPUTE THE INITIAL FOR TARGETING WHEN IZERO = 6 OR 7

CALLING SEQUENCE: CALL HZERIT(IZERO,RE,DI,RL,TUDR,VL)

ARGUMENTS:

IZERO	I	INITIAL CONDITION FLAG
RE	I	INJECTION RADIUS VECTOR (KM)
DI	2	JULIAN DATE OF INJECTION
RL	I	RADIUS VECTOR TO LIBRATION POINT (KM)
TDUR	I	FLIGHT TIME (DAYS)
VL	O	VELOCITY VECTOR AT LIBRATION POINT

LOCAL SYMBOLS:

ALP	ANGLE BETWEEN RL AND VL (RAD)
VLM	MAGNITUDE OF VL
TH	TRANSFER ANGLE (RAD)
SMA	SEMI-MAJOR AXIS OF TRANSFER ORBIT
E	ECCENTRICITY OF TRANSFER ORBIT
P	SEMI-LATUD RECTUM OF TRANSFER ORBIT
NTAB	NUMBER OF LOOKUP TABLES
KTAB	VALID LOOKUP TABLE
NP	NUMBER OF POINTS IN LOOKUP TABLE MINUS ONE
ZAXIS	PSEUDO-POLE VECTOR
FTAB	LOOKUP TABLES
FLIM	LOOKUP TABLE LIMITS
NLIM	NUMBER OF POINTS IN LOOKUP TABLES

SUBROUTINES REQUIRED:

DUXV	LAMBRT
DANGMD	LOOP
DVCOMB	DANGV2

COMMON USED:

PI	RETRO	SMU
RPD	ATRY	
SPD	NFR	

HZERIT FLOWCHART

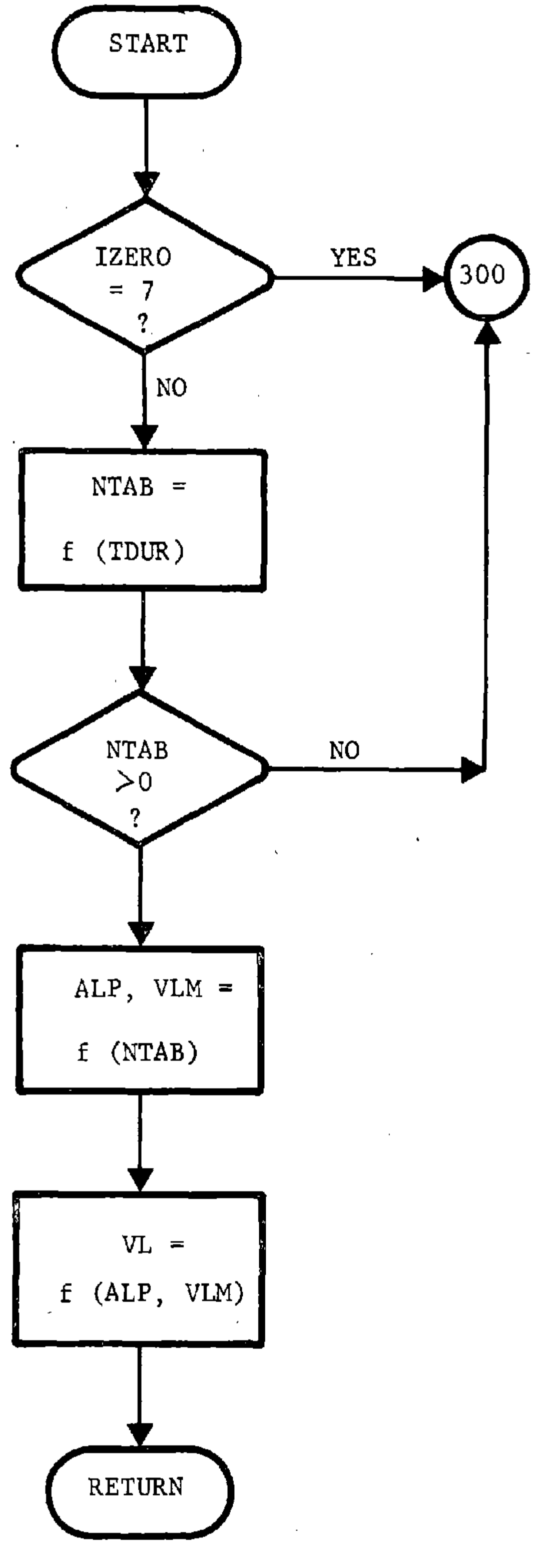
Are initial conditions generated from Lamberts solution

Find correct table for this flight time

Does flight time lie within any table

Interpolate table for velocity magnitude and rotation angle

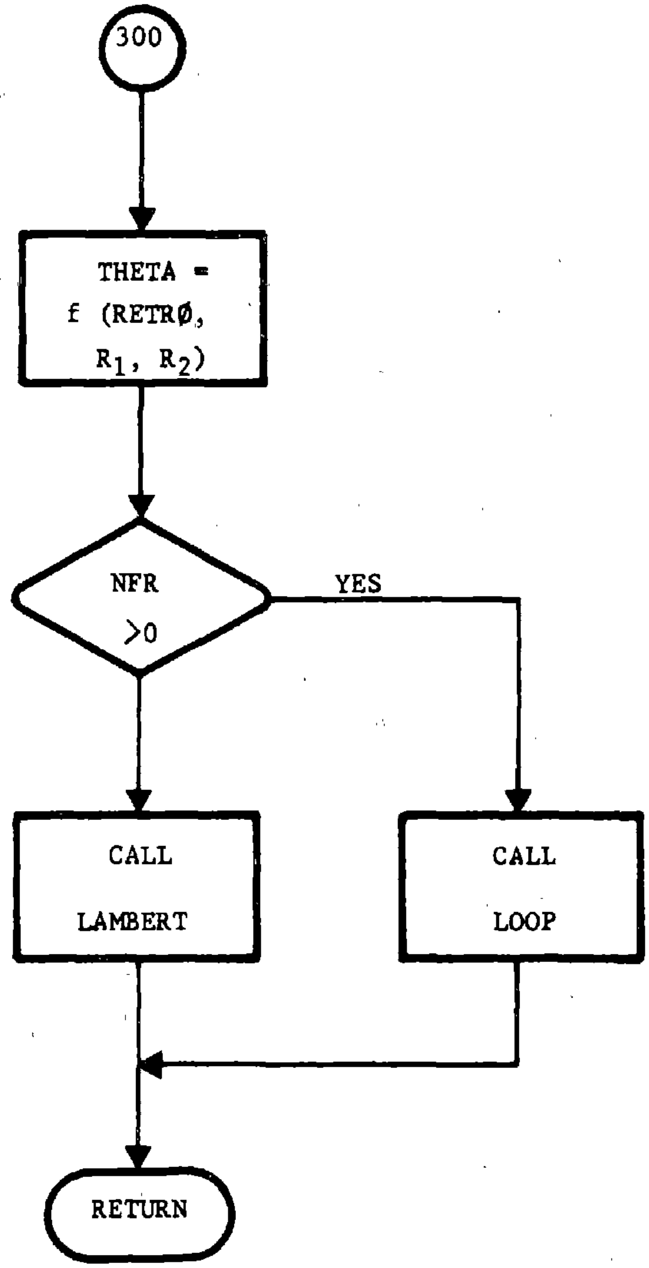
Compute initial guess of libration point velocity



Compute transfer angle

Check for transfer greater than one full revolution

Compute velocity at libration point from Lamberts solution



SUBROUTINE INITLC

PURPOSE: TO INITIALIZE CONSTANTS

ARGUMENTS:
NONE

LOCAL SYMBOLS:

PLN	CHARACTER STRING OF PLANET NAMES

SUBROUTINES REQUIRED
DSHIFT

COMMON COMPUTED:

PI	SMU
TWOPI	RSOI
RPD	RP
SPD	MONTH
XKMPAU	PLANET

SUBROUTINE JACOBI

PURPOSE: TRANSFORMATION OF A REAL SYMMETRIC MATRIX TO DIAGONAL FORM BY A SUCCESSION OF PLANE ROTATIONS TO ANNIHILATE THE OFF-DIAGONAL ELEMENTS AND SUBSEQUENT COMPUTATION OF THE EIGENVALUES AND EIGENVECTORS OF THAT MATRIX

CALLING SEQUENCE: CALL JACOBI(A,W2,V,N,FOD)

ARGUMENT:

A	I	MATRIX TO BE DIAGONALIZED (WILL BE DESTROYED)
W2	O	VECTOR OF EIGENVALUES (LENGTH N)
V	O	MATRIX OF EIGENVECTORS (N BY N DIMENSION)
N	I	DIMENSION OF SQUARE MATRIX A
FOD	I	FINAL OFF-DIAGONAL ANNIHILATION VALUE

SUBROUTINES SUPPORTED: EIGHY GUISIM GUISS PRESIM SETEVN GUIDM GUID PRED

LOCAL SYMBOLS:

AIIP	INTERMEDIATE VARIABLE
AIPIP	INTERMEDIATE VARIABLE-A(IPIP)
AIPJP	INTERMEDIATE VARIABLE-A(IPJP)
AJPJP	INTERMEDIATE VARIABLE-A(JPJP)
CS	INTERMEDIATE VARIABLE
DEL	DIFFERENCE IN ELEMENTS OF A
IREDO	COUNTER
KR	DIMENSION OF A
KRP1	KR + 1
NM1	N - 1
RAD	INTERMEDIATE VARIABLE
SN	INTERMEDIATE VARIABLE
TN	INTERMEDIATE VARIABLE
Y1	LARGEST OFF-DIAGONAL ELEMENT
VIIP	INTERMEDIATE VARIABLE

COMMON USED: ONE TWO ZERO

JACOBI Analysis

The Jacobi method subjects a real, symmetric matrix A to a sequence of transformations based on a rotation matrix:

$$O_K = \begin{bmatrix} \cos\phi_K & -\sin\phi_K \\ \sin\phi_K & \cos\phi_K \end{bmatrix}$$

where all other elements of the rotation matrix are identical with the unit matrix. After n multiplications A is transformed into:

$$A' = O_N^{-1} \dots O_1^{-1} A\, O_1 \dots O_N$$

If ϕ_K is chosen at each step to make a pair of off-diagonal elements zero, then A' will approach diagonal form with the eigenvalues on the diagonal. The columns of $O_1 O_2 \dots O_N$ correspond to the eigenvectors of A.

The angle of rotation ϕ is chosen in the following way. If the four entries of O_K are in (i,i), (i,j), (j,i) and (j,j) then the corresponding elements of $O_1^{-1} A O_1$ are

$$b_{ii} = a_{ii}\cos^2\phi + 2a_{ij}\sin\phi\cos\phi + a_{jj}\sin^2\phi$$

$$b_{ij} = b_{ji} = (a_{jj} - a_{ii})\sin\phi\cos\phi + a_{ij}(\cos^2\phi - \sin^2\phi)$$

$$b_{jj} = a_{ii}\sin^2\phi - 2a_{ij}\sin\phi\cos\phi + a_{jj}\cos^2\phi$$

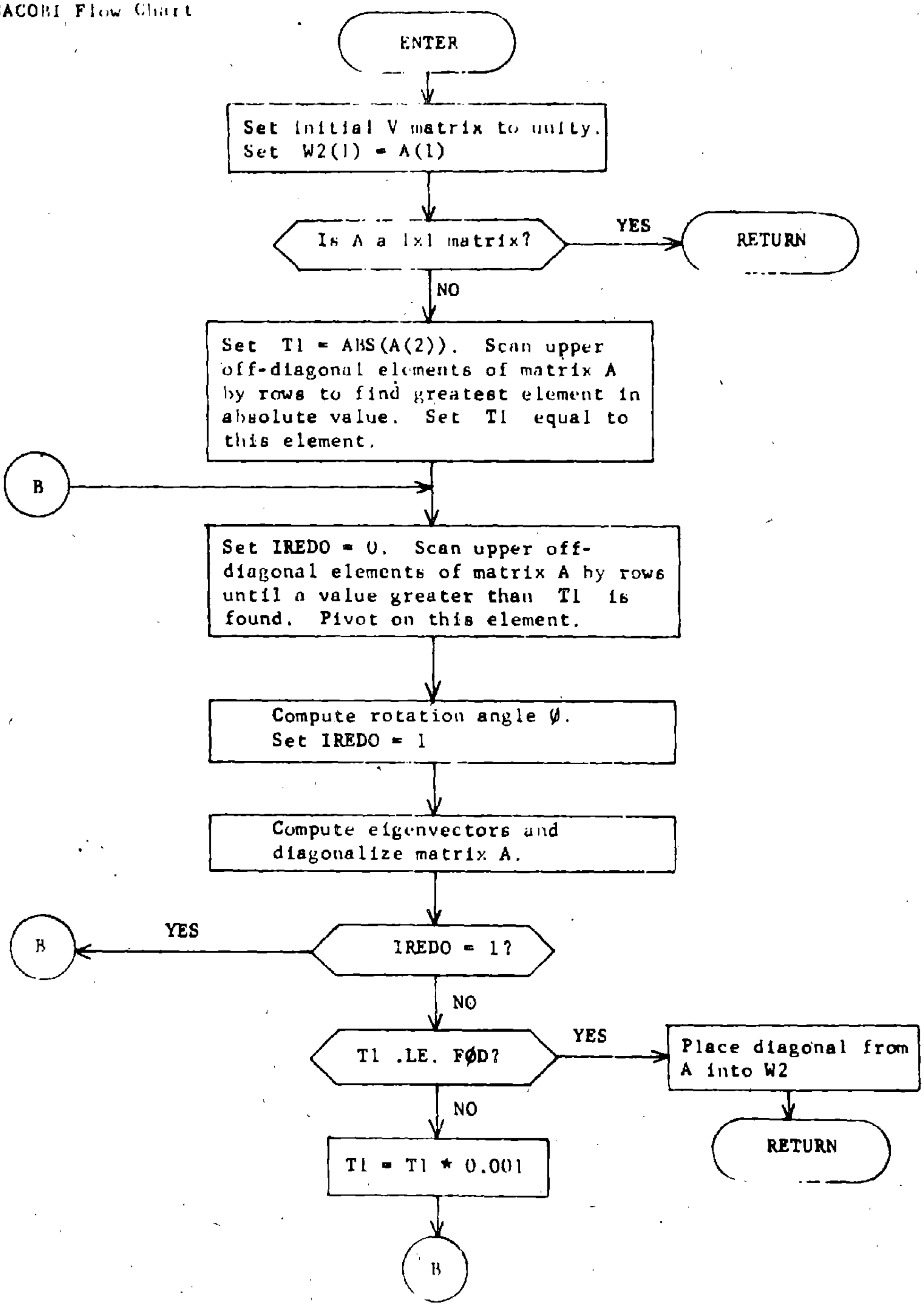
If ϕ is chosen so that $\tan 2\phi = 2a_{ij}/(a_{ii} - a_{jj})$ then

$$b_{ij} = b_{ji} = 0$$

Each multiplication creates a new pair of zeros but will introduce a non-zero contribution to positions zeroed out on previous steps. However, successive matrices of the form $O_2^{-1} O_1^{-1} A\, O_1 O_2$ will approach the required diagonal form.

Reference: Scheid, Frances: Theory and Problems of Numerical Analysis, McGraw-Hill Book Company, Inc., New York, 1968.

JACOBI Flow Chart



SUBROUTINE LAMBRT

PURPOSE: TO SOLVE LAMBERTS PROBLEM FOR TRANSFERS LESS THAN TWO PI

CALLING SEQUENCE: CALL LABRT(R1,R2,TS,THETA,XMU,V1,,V2,A,E,P)

ARGUMENTS:

R1	I	POSITION VECTOR AT DEPARTURE POINT
R2	I	POSITION VECTOR AT ARRIVAL POINT
TS	I	TRANSFER TIME
THETA	I	TRANSFER ANGLE (RAD)
XMU	I	GRAVITATIONAL CONSTANT OF CENTRAL BODY
V1	O	VELOCITY VECTOR AT DEPARTURE POINT
V2	O	VELOCITY VECTOR AT ARRIVAL POINT
A	O	SEMI-MAJOR AXIS OF TRANSFER ORBIT
E	O	ECCENTRICITY OF TRANSFER ORBIT
P	O	SEMI-LATUS RECTUM OF TRANSFER ORBIT

LOCAL SYMBOLS:

R1M	MAGNITUDE OF R1
R2M	MAGNITUDE OF R2
UR1	UNIT VECTOR IN DIRECTION OF R1
UR2	UNIT VECTOR IN DIRECTION OF R2
C	R2 - R1
UC	UNIT VECTOR IN DIRECTION OF C
CM	MAGNITUDE OF C
TPARA	PARABOLIC TIME OF FLIGHT

SUBROUTINES REQUIRED:

DABSV DANGMD
ACOSH DVCOMB
DSVECT
DVSDIV

COMMON USED:

PI
TWOPI
RPD

COMMON COMPUTED:

KWIT

LAMBRT Analysis

LAMBRT solves Lambert's problem for transfer angles less than 360°.

Given:
1) Initial and final position vectors ($\underline{R}_1$ and $\underline{R}_2$)
2) The time of flight (t_f)
3) The transfer angles (θ)

The initial and final velocity vectors ($\underline{V}_1$ and $\underline{V}_2$), for a two-body conic trajectory connecting these points are found.

The problem is solved using the following steps:

$$\underline{C} = \underline{R}_2 - \underline{R}_1$$

$$s = (R_1 + R_2 + C)/2.0$$

$$t_p = \frac{\sqrt{2}}{3} (s^{3/2} - g_1 (s - C)^{3/2})$$

where $g_1 = \text{sign} (\pi^2 - \theta^2)$

and $0 < \theta < 2\pi$

For $t_f > t_p$, the conic is an ellipse and the following transcendental equation must be solved for λ

$$\sqrt{\mu}\, t_f = \left(\frac{s}{1-\cos\lambda} \right) (\lambda - \sin\lambda) - g_1 (B - \sin B)$$

where:

$$s (1 - \cos B) = (s - c)(1 - \cos\lambda)$$

and

$$0 \leq \lambda \leq 2\pi$$

$$0 \leq A \leq \lambda$$

$$0 \leq B \leq \lambda$$

μ = gravitational constant of central body

The semi major axis of the ellipse may now be calculated as:

$$a = s / (1 - \cos\lambda)$$

Also calculate: $g_2 = \text{sign} (\pi^2 - \lambda^2)$

For $t_f < t_p$, the conic is an hyperbola, and r must be solved for in the transcendental equation:

$$\sqrt{\mu}\ t_f = \left(\frac{s}{\cosh r - 1}\right)^{3/2} (\sinh r - r) - g_1 (\sinh \delta - \delta)$$

where:

$$s (\cosh \delta - 1) = (s - C)(\cosh r - 1)$$

and

$$0 \leq \delta \leq r \leq \infty$$

The semi major axis of the transfer hyperbola is

$$a = s/(1 - \cosh r)$$

and set $g_2 = +1$

For both elliptical and hyperbolic transfer orbits the initial and final velocities are now calculated from the following equations:

$$A = g_1 \sqrt{\frac{1}{s - C} - \frac{1}{2a}}$$

$$B = g_2 \sqrt{\frac{1}{s} - \frac{1}{2a}}$$

$$V_c = \sqrt{\frac{\mu}{2}} (A + B)$$

$$V_p = \sqrt{\frac{\mu}{2}} (A - B)$$

$$\underline{V}_1 = V_c \hat{C} + V_p \hat{R}_1$$

$$\underline{V}_2 = V_c \hat{C} - V_p \hat{R}_2$$

For the derivation of these equations, the reader is referred to Astronautical Guidance by R. H. Batten.

SUBROUTINE LOOP

PURPOSE: TO SOLVE LAMBERTS PROBLEM FOR TRANSFERS GREATER THAN TWO PI

CALLING SEQUENCE: CALL LOOP(R1,R2,TS,THETA,XMU,ATRY,NFR,V1,V2,A,E,P)

ARGUMENTS:

R1	I	POSITION VECTOR AT DEPARTURE POINT
NFR	I	NUMBER OF FULL REVOLUTIONS BEFORE ENCOUNTER
NFR	I	NUMBER OF FULL REVOLUTIONS BEFORE ENCOUNTER
ATRY	I	INITIAL GUESS

LOCAL SYMBOLS:

RM1	MAGNITUDE OF R1
RM2	MAGNITUDE OF R2
CM	MAGNITUDE OF C
C	R2 - R1
CTH1	COSINE OF TRUE ANOMALY AT POSITION 1
CTH2	COSINE OF TRUE ANOMALY AT POSITION 2
STH1	SINE OF TRUE ANOMALY AT POSITION 1
STH2	SINE OF TRUE ANOMALY AT POSITION 2
THSC1	TRUE ANOMALY AT POSITION 1
THSC2	TRUE ANOMALY AT POSITION 2
VM2	MAGNITUDE OF V1
VM1	MAGNITUDE OF V2

SUBROUTINES REQUIRED:

DSVECT	DABSV	SL0020
DUNIT	DUXV	DVSMLT
DVCOMB	DANDMD	

COMMON USED:

PI
RPD

LOOP Analysis

LOOP solves Lambert's problem for transfer angles greater than 360°.

Given: 1) Initial and final position vectors

$$\underline{R}_1 \text{ and } \underline{R}_2$$

2) The time of flight (t_f)

3) The transfer angle (θ_N)

The initial and final velocity vectors ($\underline{V}_1$ and $\underline{V}_2$) for a two-body elliptical conic connecting these points are found.

The problem is solved using the following step.

$$\underline{C} = \underline{R}_2 - \underline{R}_1$$

$$s = (R_1 + R_2 + C)/2.0$$

$$Q = \sqrt{\frac{(R_1 + R_2)}{s}} \cos (\theta/2)$$

where $\theta = (\theta_N) \bmod 2\pi$

The following sets of equations must be solved for X by interating

$$y^3 \sqrt{\frac{\mu}{s}} \frac{t_f}{s} = (m\pi + \lambda - h)$$

$$E = X^2 - 1$$

$$y = (-E)^{\frac{1}{2}}$$

$$K = Q^2$$

$$z = (1 + KE)^{\frac{1}{2}}$$

$$f = y (z-Qx)$$

$$g = xz-QE$$

$$\lambda = \tan^{-1} (f/g)$$

$$d = m\pi + \lambda$$

The orbital elements are then

$$a = \frac{\frac{1}{2} s}{y^2}$$

$$r_d = \frac{(2\mu s)^{\frac{1}{2}} \quad Qz \; (s - R_1 \quad) - x \; (s - \quad R_2 \quad)}{c \quad R_1}$$

$$e^2 = (1 - \frac{R_1}{a})^2 + \frac{(R_1 \quad r_d)^2}{(\frac{\mu}{a})}$$

For the derivation of the previous equations, the reader is refered to NASA Technical Note D-5368 (A Unified Form of Lambert's Theorem, by E. R. Lancaster and R. C. Blanchard).

The initial and final velocities are now calculated using the method:

$$p = a \; (1 - e^2)$$

$$\cos N_1 = (P - \quad R_1 \quad)/(e \quad R_1 \quad)$$

$$\cos N_2 = (P - \quad R_2 \quad)/(e \quad R_2 \quad)$$

$$\sin N_1 = (\cos N_1 \cos \theta - \cos N_2)/\sin \theta$$

$$N_2 = (N_1 + \theta) \bmod 2\pi$$

$$\cos \Gamma_1 = \left(\frac{aP}{R_1 \quad (2a - \quad R_1 \quad)} \right)^{\frac{1}{2}}$$

$$\sin \Gamma_1 = e * \sin N_1 \; / \; (1 + 2e \cos N_1 + e^2)^{\frac{1}{2}}$$

$$V1^2 = \mu\left(\frac{2}{R_1} - \frac{1}{a}\right)$$

$$\cos \Gamma_2 = \left(\frac{aP}{r_2 \quad (2a - \quad R_2)} \right)^{\frac{1}{2}}$$

$$\sin \Gamma_2 = e + \sin N_2/(1 + 2e \cos N_2 + e^2)^{\frac{1}{2}}$$

$$V2^2 = \mu \left(\frac{2}{R_2} - \frac{1}{a} \right)$$

$$\hat{W} = g_1 \; (\underline{R}_1 \times \underline{R}_2)/ \quad |(\underline{R}_1 \times \underline{R}_2)|$$

where $g_1 = + 1.0 \qquad 0 \lesseqgtr \theta < \pi$

and $g_1 = - 1.0 \qquad \pi \lesseqgtr \theta < 2\pi$

$$\hat{U}_1 = \hat{W} \times \hat{R}_1$$

$$\hat{U}_2 = \hat{W} \times \hat{R}_2$$

$$\underline{V}_1 = V1\ (\sin\ \Gamma_1\ \widehat{R}_1 + \cos\ \Gamma_1\ \widehat{U}_1\ )$$

$$\underline{V}_2 = V2\ (\sin\ \Gamma_2\ \widehat{R}_2 + \cos\ \Gamma_2\ \widehat{U}_2\ )$$

ROUTINE MAIN

PURPOSE: ENTRY POINT TO PROGRAM NOMNAL

LOCAL SYMBOLS:

OPTION	CHARACTER STRING READ FROM INPUT
HALO	CHARACTER STRING 'HALO'

SUBROUTINES REQUIRED:

INITLC
HPRELM
HTRJTY
PRELIM
HGIDNS
TRJTRY
GIDANS

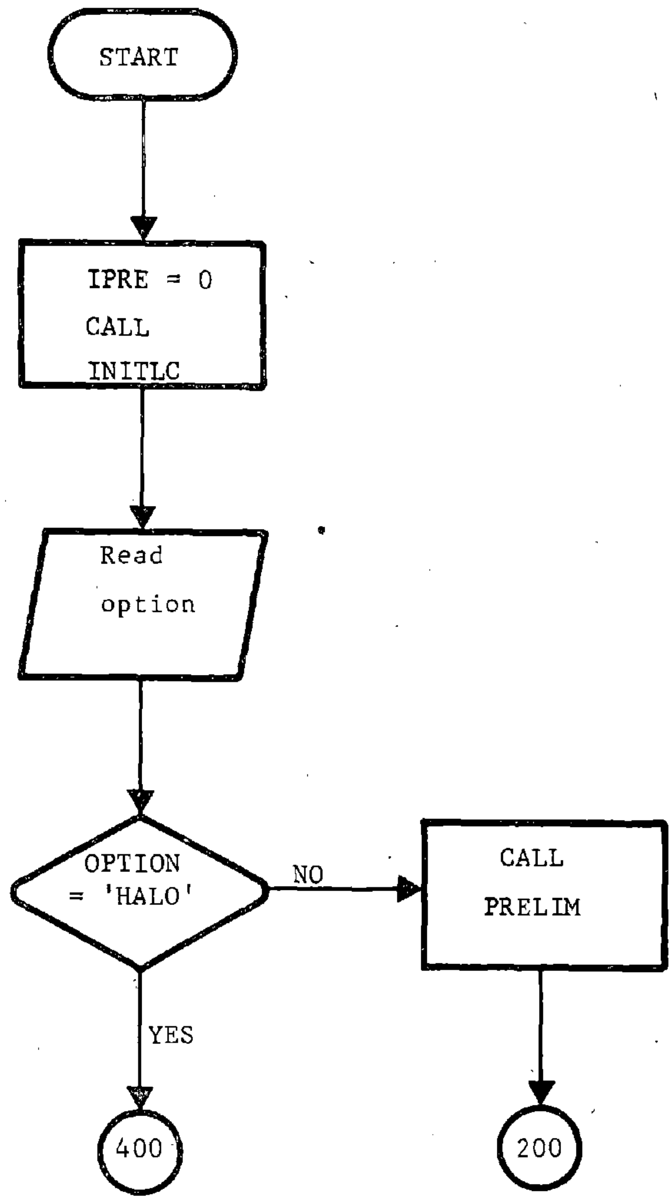
COMMON USED:

KWIT

COMMON COMPUTED:

IPRE

MAIN FLOWCHART



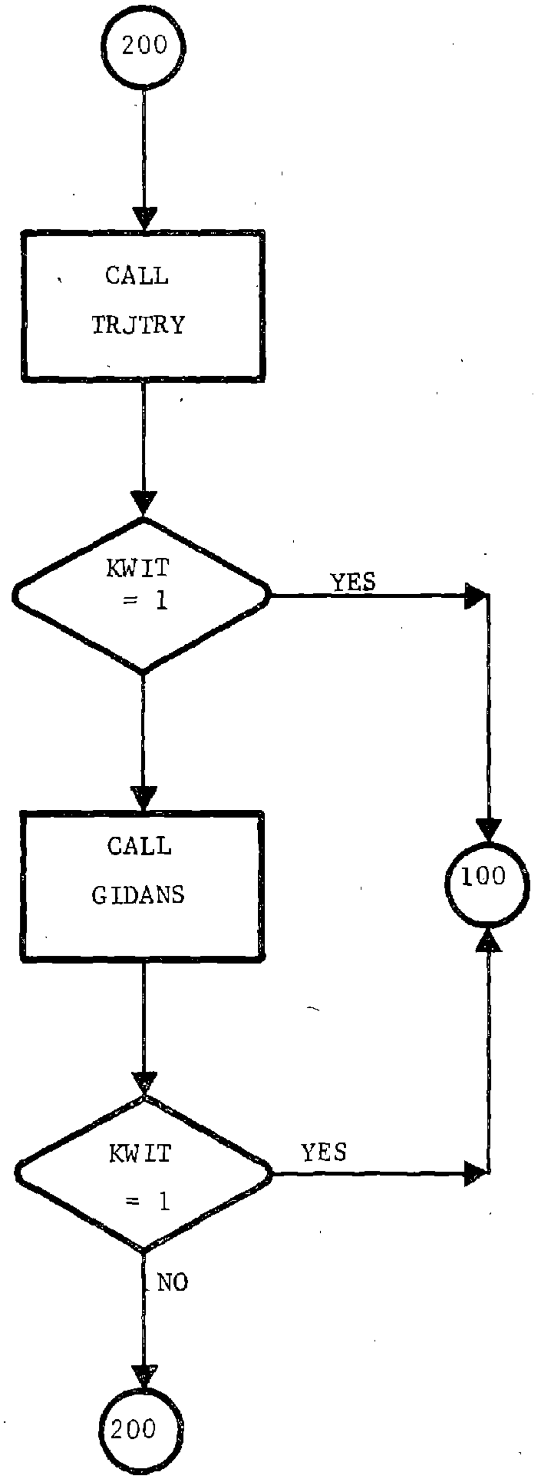
Check if halo orbit mission

Non-halo orbit trajectory generation

Case termination?

Non-halo orbit trajectory

Cast termination?



Initialize data for halo orbit mission

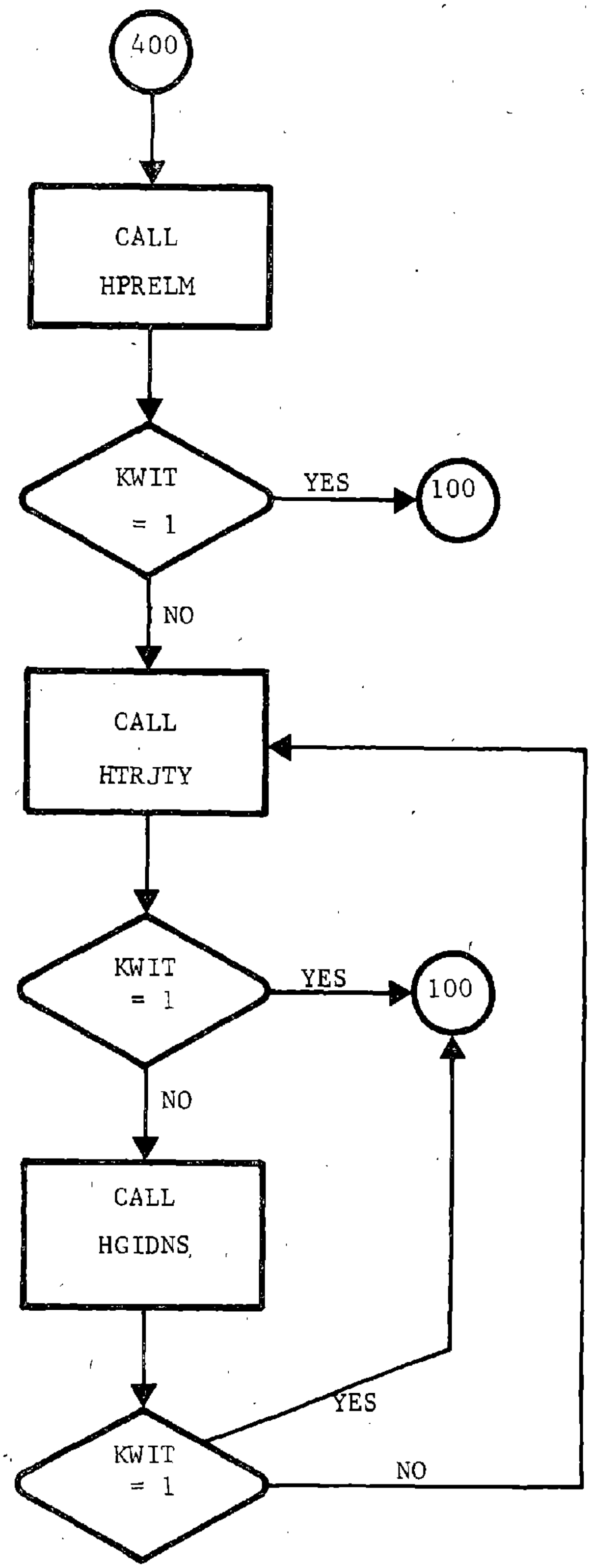
Case termination?

Generate halo orbit trajectory

Case termination?

Target halo orbit trajectory

Case termination?



SUBROUTINE MATIN

PURPOSE: TO COMPUTE THE INVERSE OF A MATRIX.

CALLING SEQUENCE: CALL MATIN(A,R,N)

ARGUMENTS	A(N,N)	I	MATRIX TO BE INVERTED
	R(N,N)	O	RESULTANT INVERSE OF MATRIX A
	N	I	DIMENSION OF A AND R

SUBROUTINES REQUIRED: NONE

LOCAL SYMBOLS:	AL	A(LL) + S (INTERMEDIATE VARIABLE)
	ALBAR	INTERMEDIATE VARIABLE
	B	INTERMEDIATE VECTOR
	DETR	INTERMEDIATE VECTOR
	G	INTERMEDIATE VECTOR
	IX	INTERMEDIATE VECTOR
	KR	DIMENSION OF A
	MIXI	INTERMEDIATE VARIABLE
	MIXJ	INTERMEDIATE VARIABLE
	MIXL	INTERMEDIATE VARIABLE
	S	INTERMEDIATE VARIABLE
	X	INTERMEDIATE VARIABLE
	XOFF	INTERMEDIATE VARIABLE

SUBROUTINE MEAN

PURPOSE: TO PROPAGATE AND UPDATE MEANS OF ACTUAL STATE AND PARAMETER DEVIATIONS AND ACTUAL STATE AND PARAMETER ESTIMATION ERRORS

CALLING SEQUENCE: CALL MEAN(EXTP,EXSTP,IFLAG,IFLAG1,NR)

ARGUMENTS: EXTP I STATE DEVIATIONS OR ESTIMATION ERRORS

EXSTP I SOLVE-FOR PARAMETER DEVIATIONS OR ESTIMATION ERRORS

IFLAG I =1 FOR UPDATE
=2 FOR PROPAGATION

IFLAG1 I =1 FOR DEVIATION MEANS
=2 FOR ESTIMATION ERROR MEANS

NR I NUMBER OF ROWS IN THE OBSERVATION MATRIX

SUBROUTINES SUPPORTED: ERRANN SETEVN PROBE GENGID PRED

LOCAL SYMBOLS: IGO INTERNALLY SET FLAG

SUM INTERMEDIATE STORAGE

ZERO VALUE 0.0

COMMON COMPUTED/USED: DUME EU EV EW EXIP EXSIP

COMMON USED: AK AL AM AN G H NDIM1 NDIM2 NDIM3 NDIM4 PHI S TXU TXW TXXS

MEAN Analysis

Subroutine MEAN propagates and updates actual estimation error means over the time interval $[t_k, t_{k+1}]$ separating twc successive measurements or events. The equations programmed in MEAN are independent of the filter algorithm employed to generate gain matrices. Gain matrices are assumed to have been computed during a prior call to subroutine GNAVM. The propagation equations programmed in MEAN are also used to propagate actual deviation means over the time interval separating two successive guidance events. The update equations, of course, are not used in this situation.

The actual estimation errors for position/velocity state, solve-for parameters, dynamic consider parameters, measurement consider parameters, and ignore parameters are defined, respectively, by the following:

$$\tilde{x}_{k+1} = \hat{x}_{k+1} - x_{k+1} \tag{1}$$

$$\tilde{x}_{s_{k+1}} = \hat{x}_{s_{k+1}} - x_{s_{k+1}} \tag{2}$$

$$\tilde{u}_{k+1} = \hat{u}_{k+1} - u_{k+1} = - u_o \tag{3}$$

$$\tilde{v}_{k+1} = \hat{v}_{k+1} - v_{k+1} = - v_o \tag{4}$$

$$\tilde{w}_{k+1} = \hat{w}_{k+1} - w_{k+1} = - w_o \tag{5}$$

where (^) indicates estimated values, and x, x_s, u_o, v_o, and w_o are the actual deviations from nominal.

Only the means of $\tilde{x}$ and $\tilde{x}_s$ are paopagated and updated since the means of u, v, and w are constant. The propagation equations are summarized

$$E\left[\tilde{x}^-_{k+1}\right] = \Phi \cdot E\left[\tilde{x}^+_k\right] + \theta_{xx_s} \cdot E\left[\tilde{x}^+_{s_k}\right] - \theta_{xu}\, \bar{u}_o - \theta_{xw}\, \bar{w}_o \tag{6}$$

$$E\left[\tilde{x}^-_{s_{k+1}}\right] = E\left[\tilde{x}^+_{s_k}\right] \tag{7}$$

where Φ, θ_{xx_s}, θ_{xu}, and θ_{xw} are state transition matrices over $[t_k, t_{k+1}]$.

Before the means of x and x_s can be updated at a measurement, the mean of the measurement residual $_{k+1}$ must first be computed using

$$E\left[\varepsilon_{k+1}\right] = - H \cdot E\left[\tilde{x}^-_{k+1}\right] - M \cdot E\left[\tilde{x}^-_{s_{k+1}}\right] + G\bar{u}_o + L\bar{v}_o + N\bar{w}_o \qquad (8)$$

where H, M, G, L, and N are observation matrix partitions.

The update equations are summarized as:

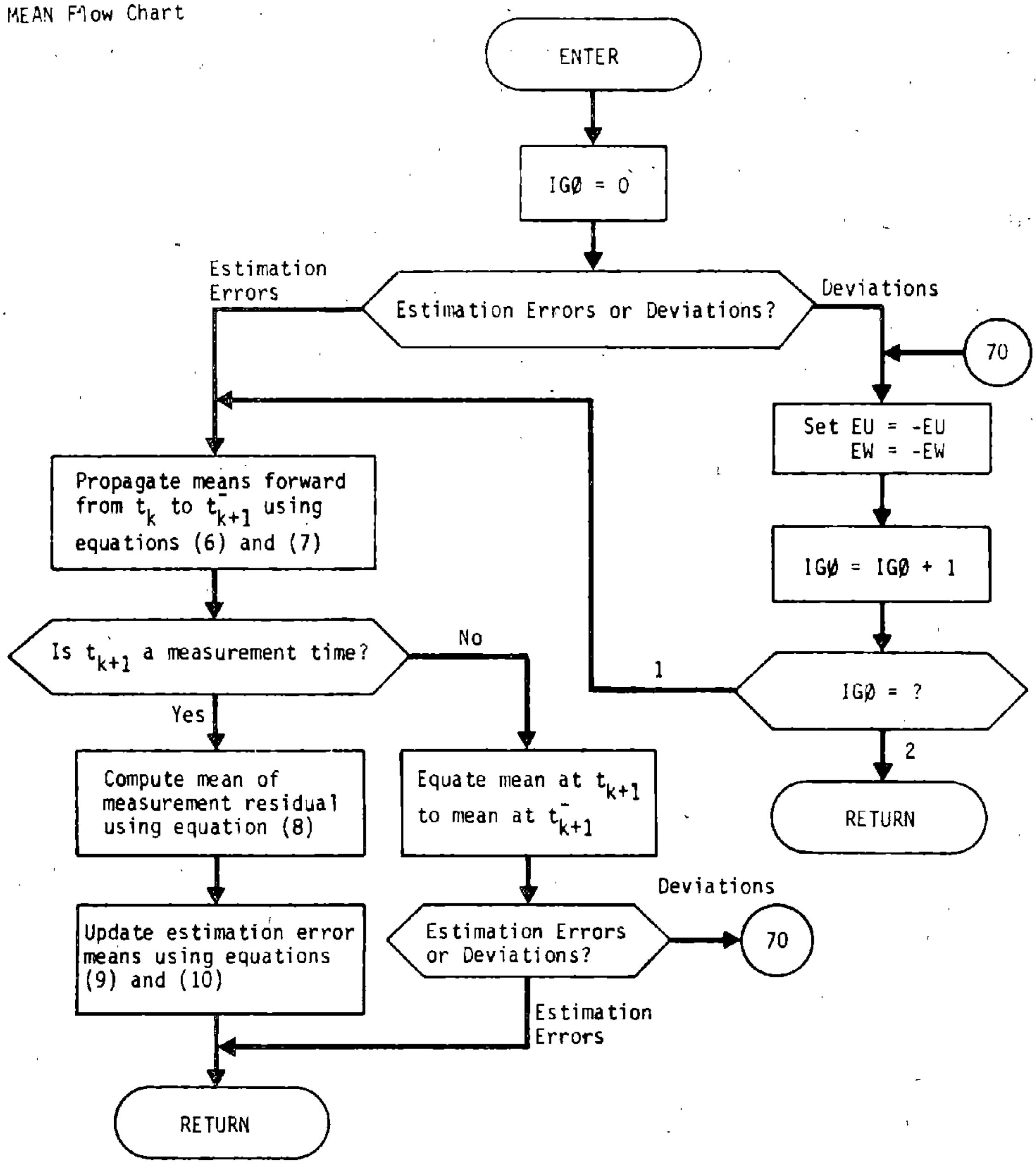
$$E\left[\tilde{x}^+_{k+1}\right] = E\left[\tilde{x}^-_{k+1}\right] + K_{k+1} \cdot E\left[\varepsilon_{k+1}\right] \qquad (9)$$

$$E\left[\tilde{x}^+_{k+1}\right] = E\left[\tilde{x}^-_{s_{k+1}}\right] + S_{k+1} \cdot E\left[\varepsilon_{k+1}\right] \qquad (10)$$

where K_{k+1} and S_{k+1} are the filter gain matrices.

To propagate actual deviation means requires that x and x_s be replaced by $\tilde{x}$ and $\tilde{x}_s$, respectively, in equations (6) and (7), and that the minus signs in equations (6) be replaced with plus signs.

MEAN Flow Chart



SUBROUTINE MENO

PURPOSE: COMPUTE ASSUMED AND ACTUAL MEASUREMENT NOISE COVARIANCE MATRICES IN THE ERROR ANALYSIS PROGRAM

CALLING SEQUENCE: CALL MENO(MMCODE,ICODE)

ARGUMENT: ICODE I INTERNAL CODE USED TO DISTINGUISH BETWEEN THE TWO ALTERNATIVES LISTED ABOVE

MMCODE I MEASUREMENT MODEL CODE

SUBROUTINES SUPPORTED: ERRANN

COMMON COMPUTED: R RPR

COMMON USED: IMNF MNCN IGMNF GMNCN

MENØ Analysis

The linearized observation equation employed by the navigation process is given by

$$\delta Y_k = H_k^A \, \delta X_k^A + n_k$$

where δY_k is the measurement deviation from the nominal measurement, H_k^A is the augmented observation matrix, δX_k^A is the augmented state deviation from the nominal augmented state, and n_k is the assumed measurement noise.

The function of subroutine MENØ is to compute the assumed measurement noise covariance matrix

$$R_k = E\left[n_k \; n_k^T\right]$$

if ICØDE = 0. The constant measurement noise variances associated with all available measurement types are stored in the vector MNCN. Subroutine MENØ selects the appropriate element from this vector to construct R_k.

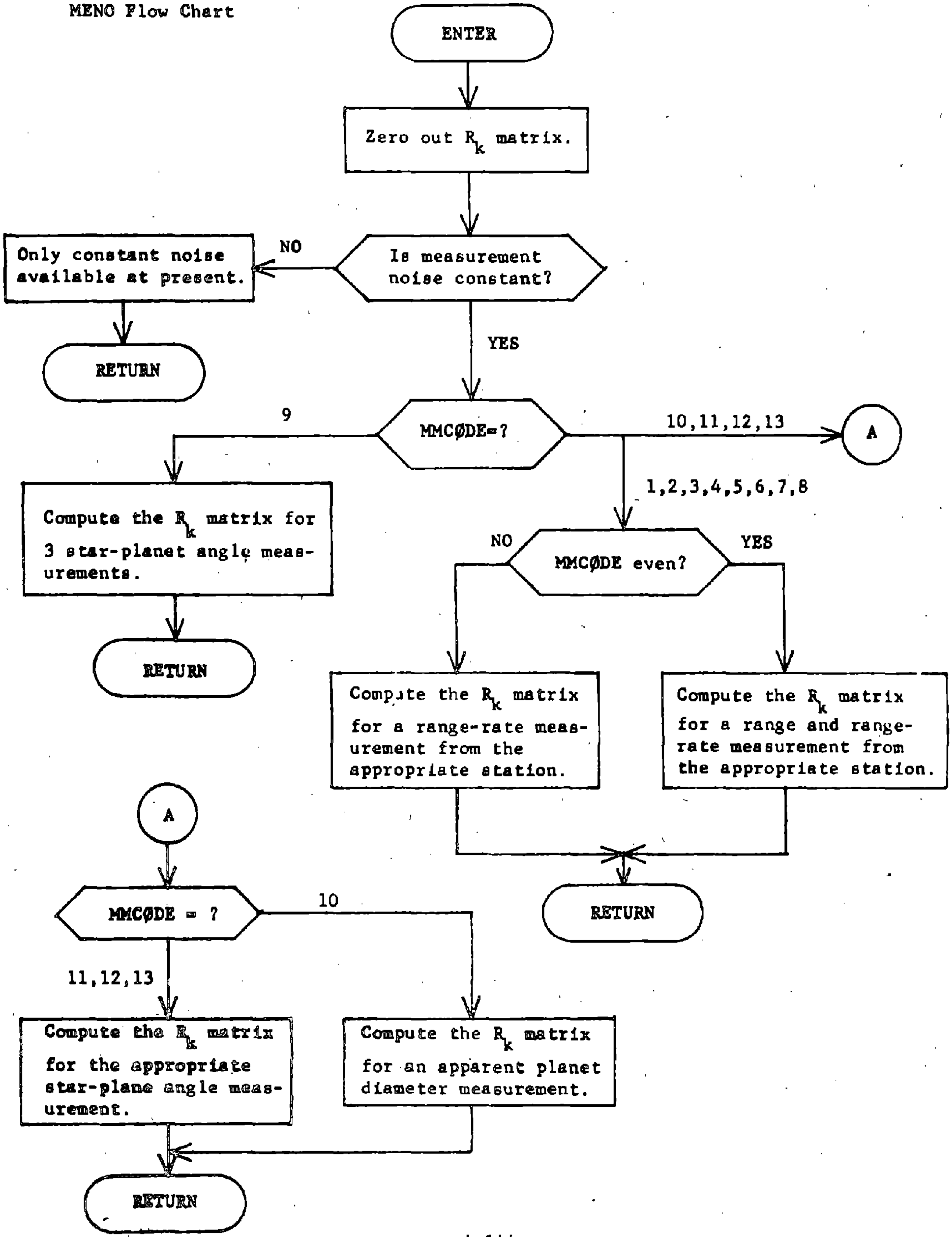
If ICØDE ≠ 0 the actual measurement noise covariance matrix

$$R_k' = E\left[n_k' \; n_k'^T\right]$$

where n_k' is the actual measurement noise, is computed instead. In this case subroutine MENØ selects the appropriate actual measurement noise variances from the vector GMNCN to construct R_k'.

The accompanying flow chart indicates the computational flow for computing R_k. An identical procedure is used to compute R_k'.

MENO Flow Chart



SUBROUTINE MOMENT

PURPOSE: TO CONVERT AN ARBITRARY NON-SQUARE 2ND MOMENT MATRIX TO THE ASSOCIATED CORRELATION MATRIX PARTITION AND PRINT IT ALSO COMPUTE AND PRINT EIGENVALUES, EIGENVECTORS,AND HYPERELLIPSOIDS

CALLING SEQUENCE: CALL MOMENT(N1,N2,EXYT,EX,EY,CORW,CORW1,ABL,I1, I2,IFLAG,IF2)

ARGUMENTS:

N1	I	NUMBER OF ROWS IN 2ND MOMENT MATRIX
N2	I	NUMBER OF COLS IN 2ND MOMENT MATRIX
EXYT	I	N1 BY N2 2ND MOMENT MATRIX OF X AND Y
EX	I	N1 VECTOR MEAN OF X
EY	I	N2 VECTOR MEAN OF Y
CORW	I	2ND MOMENT MATRIX CORRESPONDING TO VECTOR X OF DIMENSION N1
CORW1	I	2ND MOMENT MATRIX CORRESPONDING TO VECTOR Y OF DIMENSION N2
ABL	I	VECTOR OF ROW LABELS CORRESPONDING TO CORW1
I1	I	ROW INDEX MAXIMUM
I2	I	COL INDEX MAXIMUM
IFLAG	I	=0 DO NOT COMPUTE EIGENVECTORS, ETC.
IF2	I	=0 DO NOT COMPUTE STD. DEV.

SUBROUTINES SUPPORTED: GPRINT GENGID

SUBROUTINES REQUIRED: EIGHY

LOCAL SYMBOLS:

OUT	INTERMEDIATE ARRAY
PEIG	INTERMEDIATE VECTOR
ROW	INTERMEDIATE VECTOR
SQP	INTERMEDIATE VECTOR
SQP1	INTERMEDIATE VECTOR
VEIG	INTERMEDIATE VECTOR

COMMON USED: FOP FOV

MØMENT Analysis

Subroutine MØMENT transforms an arbitrary 2nd moment matrix $E[xy^T]$ into a correlation matrix and, if $x = y$, into a vector of standard deviations. The transformation consists of two steps:

1) Transform $E[xy^T]$ into the covariance matrix

$$\mathrm{cov}\,(x,y) = E[xy^T] - E[x] \cdot E[y^T];$$

2) Transform cov (x,y) into the correlation matrix having correlation coefficients

$$\rho_{ij} = \frac{\sigma_{ij}}{\sigma_i \; \sigma_j} \qquad i \neq j$$

where

$$\sigma_{ij} = E[x_i \; y_j]$$

$$\sigma_i = E[x_i^2]^{\frac{1}{2}}$$

$$\sigma_j = E[y_j^2]^{\frac{1}{2}}$$

Subroutine MØMENT writes out the correlation matrix and, if they exist, the standard deviations. Subroutine MØMENT can also compute and write out the eigenvalues, eigenvectors, and hyperellipsoid of cov (x,y) if $x = y$.

SUBROUTINE NTM

PURPOSE: TO SHIFT THE LAST STATE TRANSITION MATRIX OBTAINED FROM THE FILE INTO PHIOLD, TO CALL THE FILE READER AND OBTAIN THE NEW STATE VECTOR AND STATE TRANSITION MATRIX, PHINEW, (I.E., TO SET UP COMMON BLOCK PHISAV FOR USE BY SUBROUTINE PSIM), OR TO FLAG THE ERROR ENCOUNTERED WHILE TRYING TO READ THE FILE.

CALLING SEQUENCE: CALL NTM(RF,PHI,KSECT)

ARGUMENTS:

RF	STATE VECTOR OBTAINED FROM FILE
PHI	STATE TRANSITION MATRIX
KSECT	INDEX OF SECTION OF FILE TO BE READ =1 FOR COAST SECTION =2 FOR FINITE BURN SECTION

LOCAL SYMBOLS:

IERR	ERROR FLAG RETURNED BY FILE READER
TSEC	TIME IN SECONDS PAST THE INITIAL TIME ON THE FILE

SUBROUTINES REQUIRED: GETCOW

COMMON COMPUTED: PHINEW TOLD

COMMON USED: TRTMI TRTMB DELTM PHIOLD TM

SUBROUTINE ORBEND (ORBINT ENTRY POINT)

PURPOSE: TO WRITE A 'FINAL' RECORD TO THE SEQUENTIAL ORBIT FILE

CALLING SEQUENCE: CALL ORBEND

ARGUMENTS:

NONE

LOCAL SYMBOLS:

IFRN	LOGICAL FILE NUMBER

COMMON USED:

T	SX1	NSECTN
H	XVDD	
XDD	SV1	
SX1	SV2	

COMMON COMPUTED:

IELEVN

SUBROUTINE ORBINT

PURPOSE: TO INITIALIZE THE SEQUENTIAL ORBIT FILE WITH PARTIALS

CALLING SEQUENCE: CALL ORBINT

ARGUMENTS:
NONE

LOCAL SYMBOLS:

IFRN	LOGICAL FILE NUMBER

COMMON USED:

YMDIC	NSTATE	H	SV2
HMSIC	KSTATE	XDD	
AEINT	IPART	SX1	
SPINT	GM	SX2	
PVINT	NSECTN	XVDD	
OBLINT	T	SV1	

ORBINT Analysis

Subroutine ORBINT contains five entry points: ORBINT, ORBWRT, ORBEND, CSTART and GETCØW. The purpose of these entry points is to transmit and retrieve information on a sequential orbit that contains partials. Entry point orbint writes a 1048_{10} byte logical header record followed by a 1056_{10} byte second logical header record. This routine is called only once at the start of trajectory generation. After the header records have been written the ORBWRT logic block is executed. This block writes a 6657_{10} logical data record containing the latest 11 acceleration vectors, first and second cowell runs, current integration time, step size and section number. This logic block is repeatedly used during trajectory integration by a call through entry point ORBWRT. After the trajectory has been completely integrated a call to entry point ORBEND will write a data record of any remaining acceleration vectors and set an end-of-file mark.

Entry points CSTART and GETCOW are used to retrieve information from the sequential trajectory data file written by ORBINT/ORBWRT/ORBEND. An initial call to entry point CSTART prior to trajectory generation will read the two headon records and the first data record. Each successive call to entry point GETCOW will read successive data records (if required) and call subroutine INTP to interpolate the trajectory data for the requested time period.

SUBROUTINE ORBWRT (ORBINT ENTRY POINT)

PURPOSE: TO WRITE RECORDS TO THE SEQUENTIAL ORBIT FILE

CALLING SEQUENCE: CALL ORBWRT

ARGUMENTS:

NONE

LOCAL SYMBOLS:

IFRN	LOGICAL FILE NUMBER

COMMON USED:

T	SX2	NSECTN
H	XVDD	
XDD	SV1	
SX1	SV2	

COMMON COMPUTED:

IELEVN

SUBROUTINE PECEQ

PURPOSE: TO COMPUTE THE MATRIX DEFINING THE TRANSFORMATION FROM PLANET CENTERED ECLIPTIC COORDINATES TO PLANET CENTERED EQUATORIAL COORDINATES AS A FUNCTION OF THE PARTICULAR PLANET AND TIME.

CALLING SEQUENCE: CALL PECEQ(NP,D,ECEQ)

ARGUMENT	NP	I	CODE OF PLANET
	D	I	JULIAN DATE, EPOCH 1900, OF REFERENCE TIME
	ECEQ(3,3)	O	COORDINATE TRANSFORMATION MATRIX FROM PLANETOCENTRIC ECLIPTIC TO PLANETOCENTRIC EQUATORIAL COORDINATES

SUBROUTINES REQUIRED: EULMX ORB

LOCAL SYMBOLS:

AHCGC	COORDINATE TRANSFORMATION MATRIX FROM GEOCENTRIC ECLIPTIC TO GEOCENTRIC EQUATORIAL COORDINATES FOR EARTH - FROM ECLIPTIC TO ORBITAL PLANE COORDINATES FOR MOON
CSDECL	COSINE OF DECL
CSEOBL	COSINE OF EOBL
CSINM	COSINE OF INM
CSNOM	COSINE OF NODEM
CSRASC	COSINE OF RASC
DECL	DECLINATION OF TARGET PLANET POLE
DGTR	CONVERSION FACTOR FROM DEGREES TO RADIANS
ED	JULIAN DATE , EPOCH 4713 B.C.
EOBL	OBLIQUITY OF ECLIPTIC
INM	INDEX

NODEM	INDEX
NORM	UNIT VECTOR NORMAL TO TARGET PLANET ORBITAL PLANE
PBAR	CROSS PRODUCT OF POLE AND NORM
PMAG	MAGNITUDE OF PBAR
POLE	UNIT VECTOR ALIGNED WITH TARGET PLANET POLAR AXIS
POLMAG	MAGNITUDE OF POLE
QBARP	CROSS PRODUCT OF POLE AND PBAR
QMAG	MAGNITUDE OF QBARP
RASC	RIGHT ASCENSION OF TARGET PLANET POLE
SNDECL	SINE OF DECL
SNEOBL	SINE OF EOBL
SNINM	SINE OF INCLINATION INM
SNNDM	SINE OF NODE NDM
SNRASC	SINE OF RASC
TPRIM	BESSELIAN DATE
XI	INTERMEDIATE VALUE
XIQ	INTERMEDIATE VALUE
XL	INTERMEDIATE VALUE
XLQ	INTERMEDIATE VALUE

PECEQ Analysis

Subroutine PECEQ computes the coordinate transformation maxtrix A from planetocentric ecliptic to planetocentric equatorial coordinates for an arbitrary planet.

The derivation of A for a planet other than the earth or moon will be summarized. Matrix A is defined by

$$A = \left[\hat{X} \mid \hat{Y} \mid \hat{Z}\right]^T \tag{1}$$

where $\hat{X}$, $\hat{Y}$, and $\hat{Z}$ are unit vectors aligned with the planetocentric equatorial coordinate axes and referenced to the planetocentric ecliptic coordinate system. Unit vector $\hat{Z}$ is aligned with the planet pole. Unit vector $\hat{X}$ lies along the intersection of the of the planet equatorial and orbital planes and points at the planet vernal equinox. Unit vector $\hat{Y}$ completes the orthogonal triad and is given by

$$\hat{Y} = \hat{Z} \times \hat{X}. \tag{2}$$

It remains to obtain expressions for $\hat{X}$ and $\hat{Z}$. Let $\hat{N}$ denote the unit vector normal to the planet orbital plane, and let $\hat{P}$ denote the unit vector aligned with the planet pole. Then

$$\hat{Z} = \hat{P} \tag{3}$$

and

$$X = \frac{\hat{P} \times \hat{N}}{|\hat{P} \times \hat{N}|}. \tag{4}$$

The unit vector $\hat{N}$, referred to the ecliptic coordinate system, is given by

$$\hat{N} = \begin{bmatrix} \sin i \sin \Omega \\ -\sin i \cos \Omega \\ \cos i \end{bmatrix} \tag{5}$$

where i and Ω are the inclination and longitude of the ascending node, respectively, of the planet orbital plane. The unit vector $\hat{P}$, referred to the ecliptic system is given by

$$P = \begin{bmatrix} \cos \alpha \cos \delta \\ \cos \epsilon \sin \alpha \cos \delta + \sin \epsilon \sin \delta \\ - \sin \epsilon \sin \alpha \cos \delta + \cos \epsilon \sin \delta \end{bmatrix} \quad (6)$$

where α and δ are the right ascension and declination, respectively, of the planet pole relative to the geocentric equatorial coordinate system, and ε is the obliquity of the ecliptic. Expressions for α and δ for each planet were obtained from JPL TR 32-1306, *Constants and Related Information for Astrodynamic Calculations*, 1968, by Melbourne, *et al.*

For the earth and the moon, the transformation matrix A is written as the produce of two transformation matrices

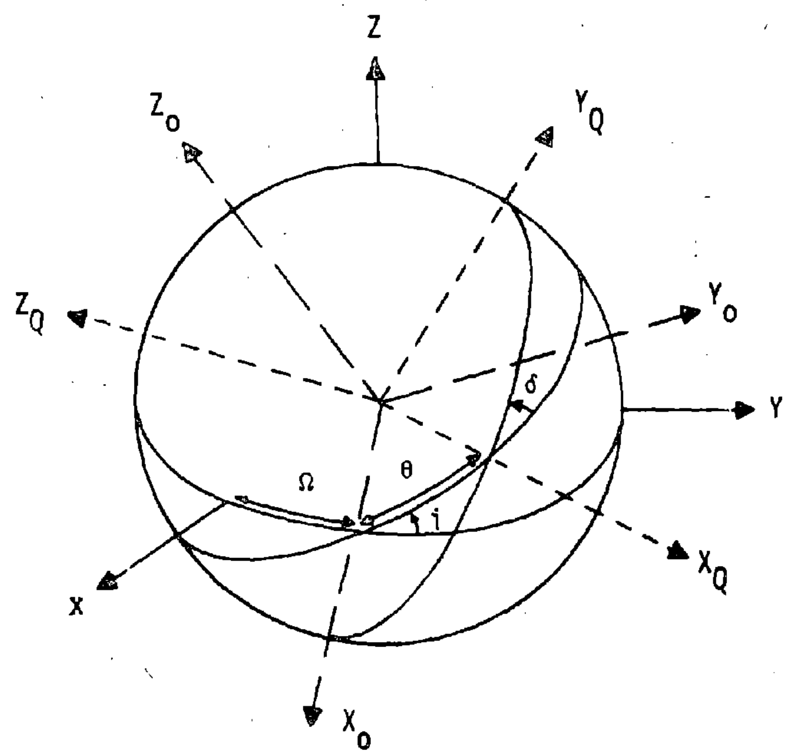
$$A = A_2 A_1. \quad (7)$$

For the earth A_2 is the identity matrix and A_1 is given by

$$A_1 = \begin{bmatrix} 1 & 0 & 0 \\ 0 & \cos \varepsilon & -\sin \varepsilon \\ 0 & \sin \varepsilon & \cos \varepsilon \end{bmatrix}. \quad (8)$$

The following figure defines the transformations A_1 and A_2, using the definitions given.

XYZ	Ecliptic coordinate axes
$X_oY_oZ_o$	Orbital plane coordinate axes
$X_QY_QZ_Q$	Moon's equatorial coordinate axes
i	Inclination of moon's orbital plane to ecliptic plane
Ω	Right ascension of moon's orbital plane to ecliptic plane
δ	Inclination of moon's equatorial to orbital plane
θ	Right ascension of moon's equatorial to orbital plane



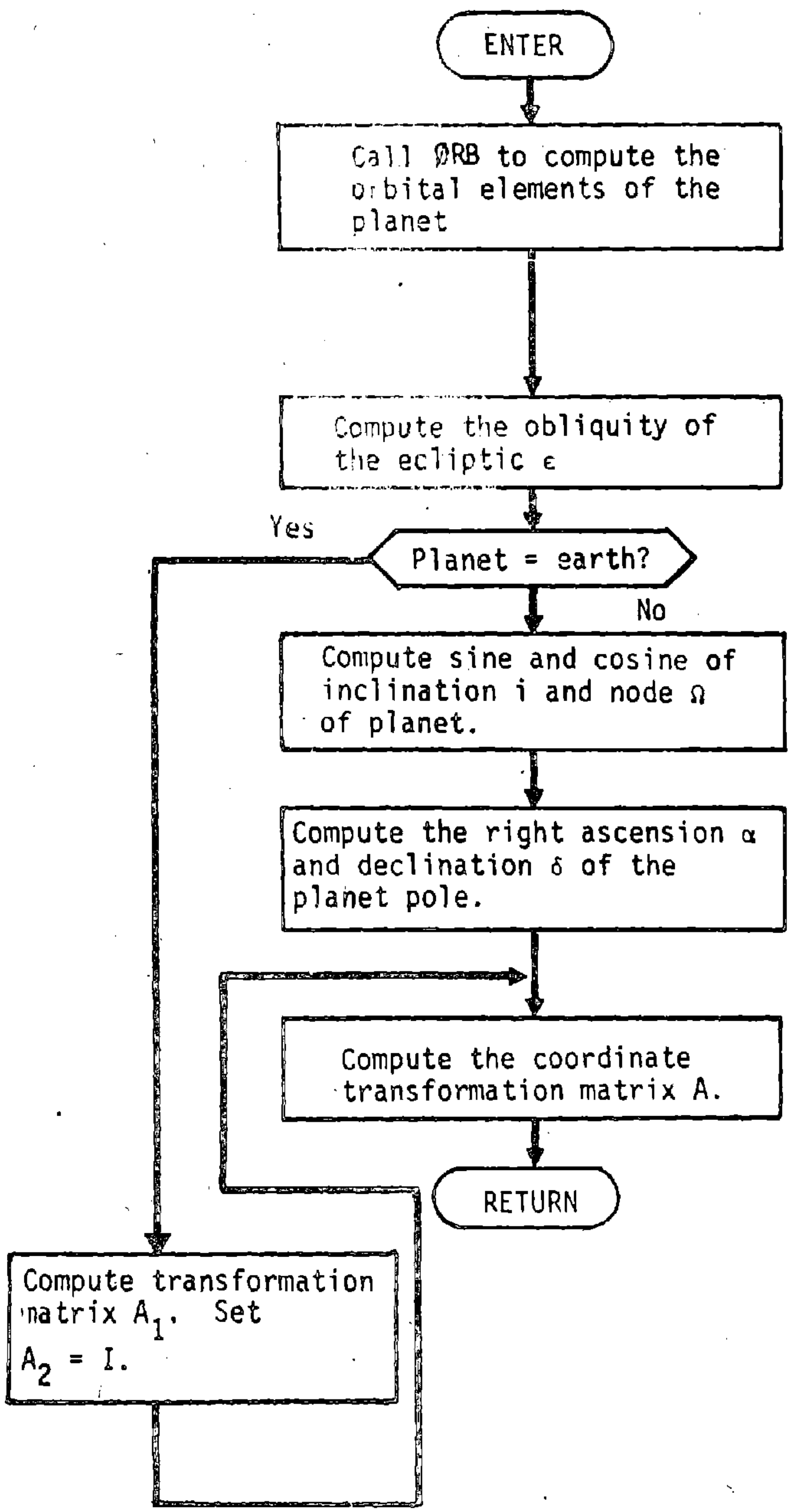
The transformation A_1 from ecliptic to orbital plane coordinates is performed by rotating about the z-axis through an angle Ω and then about the resulting x-axis through an angle i. Symbolically,

$$A_1 = (\Omega \text{ about } 3,\ i \text{ about } 1). \tag{9}$$

The transformation A_2 from orbital plane to equatorial coordinates can be written similarly as

$$A_2 = (\theta \text{ about } -3,\ \delta \text{ about } -1). \tag{10}$$

PECEQ Flow Chart



SUBROUTINE PRED

PURPOSE: CONTROL CALCULATIONS FOR A PREDICTION EVENT IN ERRAN

CALLING SEQUENCE: CALL PRED

SUBROUTINES REQUIRED: CORREL CSTART DYNO EIGHY GNAVM GPRINT MEAN NTM PSIM SAVMAT SHIFT STMPR

LOCAL SYMBOLS:

DUM2	ARRAY OF EIGENVECTORS
DUM3	ARRAY OF EIGENVALUES
DUM	INTERMEDIATE ARRAY
EGVCT	ARRAY OF EIGENVECTORS
EGVL	ARRAY OF EIGENVALUES
EXSTS	TEMPORARY STORAGE FOR EXST
EXST	TEMPORARY STORAGE FOR EXT
ICODE	INTERNAL CONTROL FLAG
IPR	TEMPORARY STORAGE FOR IPRINT
PEIG	INTERMEDIATE ARRAY
PSAVE	TEMPORARY STORAGE FOR KNOWLEDGE COVARIANCES
RF	STATE VECTOR AT TPT
RI	STATE VECTOR AT TIME OF EVENT
TPT	TIME PREDICTED TO
VEIG	MATRIX TO BE DIAGONALIZED

COMMON COMPUTED/USED: CXSU CXSV CXU CXV CXXS DELTM GCXSW GCXW IPRINT NPE P PS TRTM1 XI

COMMON USED: EM EXST EXT FOP FOV GP ISTMC NDIM1 NDIM2 NDIM3 NTMC ONE Q TPT2 U0 V0 XF

PRED Analysis

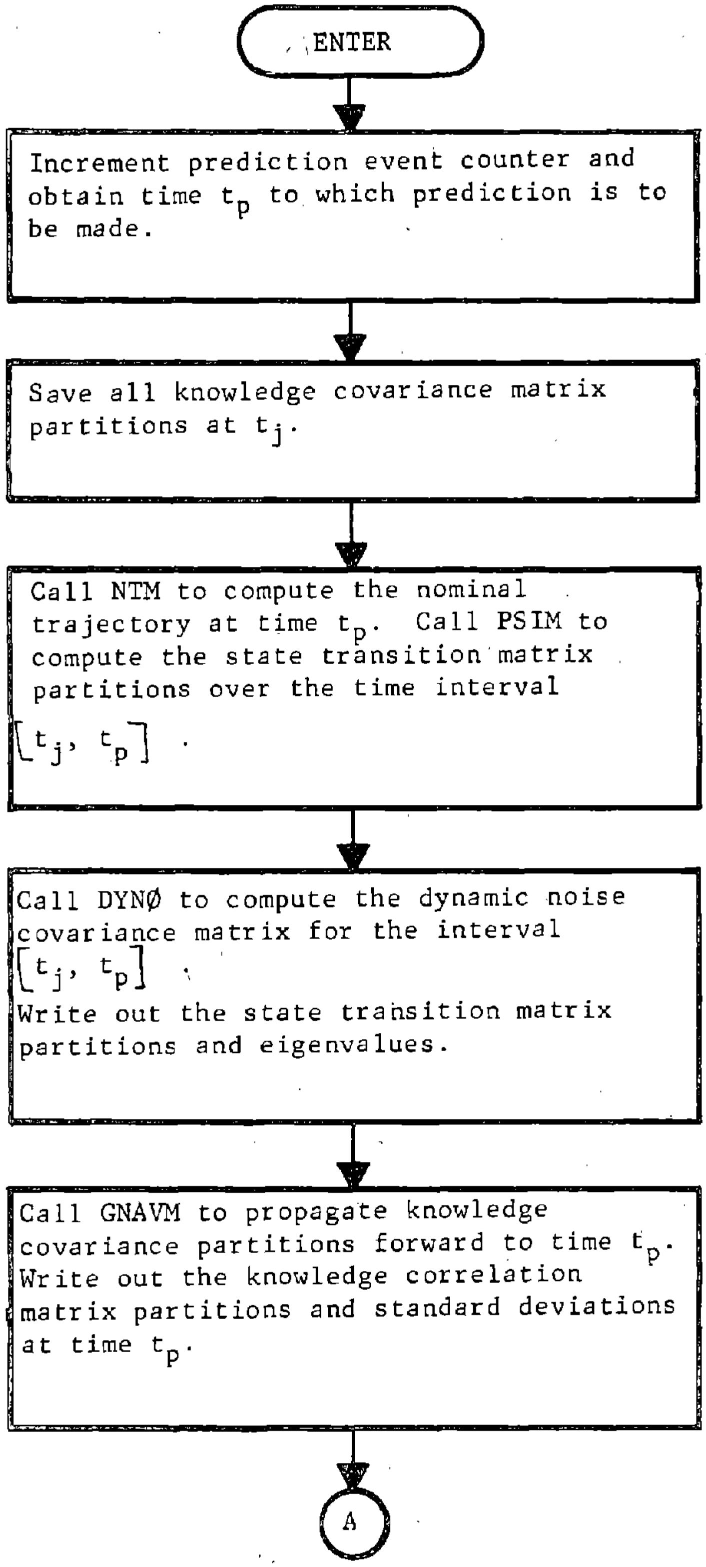
Subroutine PRED executes a prediction event in the error analysis program ERRAN. At a prediction event, the knowledge covariance partitions, and the estimated position/velocity deviations from the most recent nominal trajectory are propagated forward to t_p, the time to which the prediction is to be made. The knowledge covariance partitions are propagated using the prediction equations found in the GNAVM Analysis section. The estimate is propagated using the equation

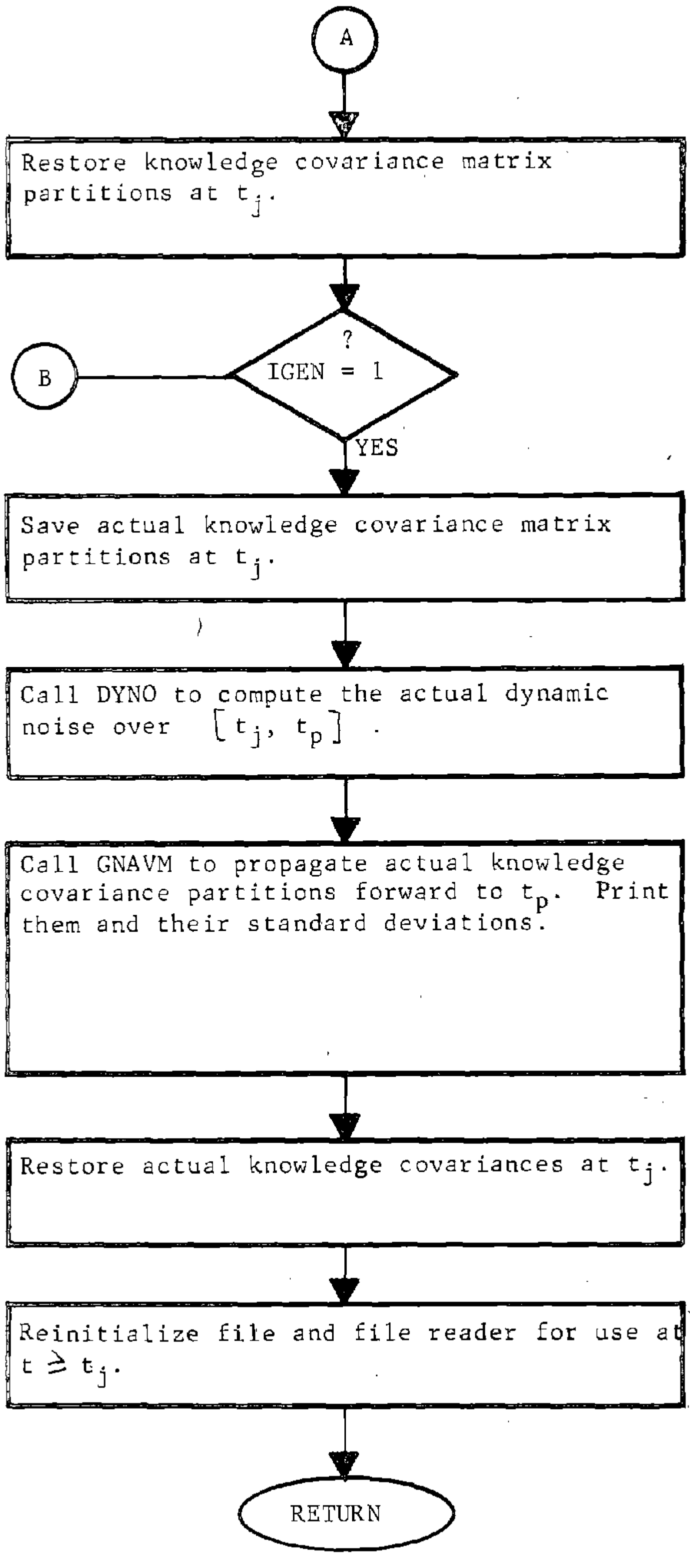
$$\delta X_p = \Phi(t_p, t) \ \delta X_j + \Theta_{xx_s} (t_p, t_j) \ \delta X_{s_j}$$

where Φ and θ_{xx_s} are the state transition matrix partitions over the time interval $\left[t_j, t_p\right]$.

The position and velocity partitions of the propagated knowledge covariance matrix are diagonalized at time t_p and the eigenvalues, and eigenvectors are computed.

PRED FLOW CHART




A
Restore knowledge covariance matrix partitions at t_j.
? IGEN = 1
B
YES
Save actual knowledge covariance matrix partitions at t_j.
Call DYNO to compute the actual dynamic noise over $[t_j, t_p]$.
Call GNAVM to propagate actual knowledge covariance partitions forward to t_p. Print them and their standard deviations.
Restore actual knowledge covariances at t_j.
Reinitialize file and file reader for use at $t \geq t_j$.
RETURN

SUBROUTINE PRELIM

PURPOSE: DUMMY LINK WITH NON HALO ORBIT OPTIONS

CALLING SEQUENCE: CALL PRELIM

SUBROUTINE PRINT3

PURPOSE: PRINT PERTINENT INFORMATION AT SPECIFIED MEASUREMENTS

CALLING SEQUENCE: CALL PRINT3(MMCODE,NR)

ARGUMENTS:	MMCODE	CODE FOR TYPE OF MEASUREMENT
	NR	NUMBER OF ROWS IN OBSERVATION MATRIX

SUBROUTINES REQUIRED: CORREL STMPR STVCPR

LOCAL SYMBOLS:	D1	HOLLERITH CONSTANT
	D2	HOLLERITH CONSTANT
	IA	STATION NUMBER
	IB	STAR NUMBER
	IONE	=1
	ITHREE	=3
	ITWO	=2
	M	INTERNAL MEASUREMENT CODE
	TRTM2	TIME OF MEASUREMENT

COMMON USED:	AK	AL	AM	AN	DELTM	H	MCNTR
	Q	R	S	TRTM1	XIG	XLAR	XSL
	XU	XV					

SUBROUTINE PSIM

PURPOSE: TO INVERT A 6X6 STATE TRANSITION MATRIX (USING ITS SYMPLECTIC CHARACTER) FROM T1 TO T2, AND MULTIPLY A STATE TRANSITION MATRIX FROM T1 TO T3 BY THAT INVERSE, AND THEREBY OBTAIN THE STATE TRANSITION MATRIX (STM) FROM T2 TO T3.

CALLING SEQUENCE: CALL PSIM(P31,P21,P32)

ARGUMENTS:

P31	STM FROM T1 TO T3
P21	STM FROM T1 TO T2
P32	STM FROM T2 TO T3

SUBROUTINES REQUIRED: NONE

COMMON USED: NONE

SUBROUTINE PSTART

PURPOSE: TO INITIALIZE THE STATE PARTIAL MATRIX

CALLING SEQUENCE: CALL PSTART

ARGUMENTS:

NONE

LOCAL SYMBOLS:

AB	MEAN RADIUS OF CENTRAL BODY

COMMON USED:

IND(1)	IBURN
ICENT	IND(16)
X	
XD	
GM	

SUBROUTINES REQUIRED:

ELEM
POLAR
PARTE

COMMON COMPUTED:

ELEMQ	NEQ
ORBEL	XV
SPHCOQ	XVD
PVINT	PXPD
AEINT	
SPINT	
OBLINT	

SUBROUTINE SAVMAT

PURPOSE: TO STORE THE 834 VALUES OF ARRAY P IN ARRAY P1

CALLING SEQUENCE: CALL SAVMAT(P,P1)

ARGUMENTS: P   ARRAY TO BE SAVED
            P1   STORAGE ARRAY

SUBROUTINE SCHED

PURPOSE: TO DETERMINE WHAT TYPE OF MEASUREMENT IS TO BE TAKEN NEXT AND AT WHAT TIME IT WILL OCCUR.

CALLING SEQUENCE: CALL SCHED(T1,T2,MMCODE)

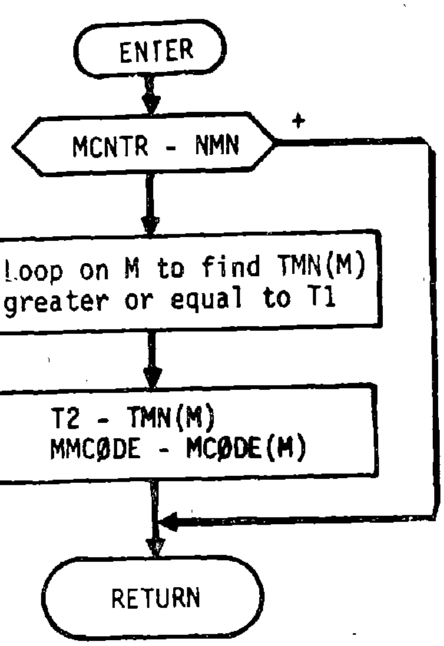
ARGUMENT:	MMCODE	O	MEASUREMENT MODEL CODE
	T1	I	PRESENT TRAJECTORY TIME
	T2	O	TRAJECTORY TIME AT WHICH THE NEXT MEASUREMENT OCCURS

SUBROUTINES SUPPORTED: ERRAN

LOCAL SYMBOLS: M INDEX

COMMON USED: MCNTR MCODE NMN TMN MCNTRP NMNP TMN1 MCODE1 TMN2 MCODE2

SCHED Flow Chart



SUBROUTINE SETEVN

PURPOSE: CONTROL COMPUTATIONS COMMON TO ALL EVENTS IN ERRAN

CALLING SEQUENCE: CALL SETEVN(RI,TEVN,NCODE)

ARGUMENTS:	RI	STATE OF TRTM1
	TEVN	TIME OF EVENT
	NCODE	TYPE OF EVENT

SUBROUTINES REQUIRED:	CORREL	DYNO	EIGHY	GNAVM	GPRINT	MEAN
	NTM	PSIM	STMPR	STVCPR		

LOCAL SYMBOLS:	EGVCT	ARRAY OF EIGENVECTORS
	EGVL	CORRESPONDING ARRAY OF EIGENVALUES
	EXTIJ	INTERMEDIATE VARIABLE
	OUT	SQUARE ROOTS OF EIGENVALUES
	PEIG	INTERMEDIATE ARRAY
	RF	STATE VECTOR AT EVENT TIME
	VEIG	MATRIX TO BE DIAGONALIZED

COMMON COMPUTED/USED: DELTM TRTM1 XF

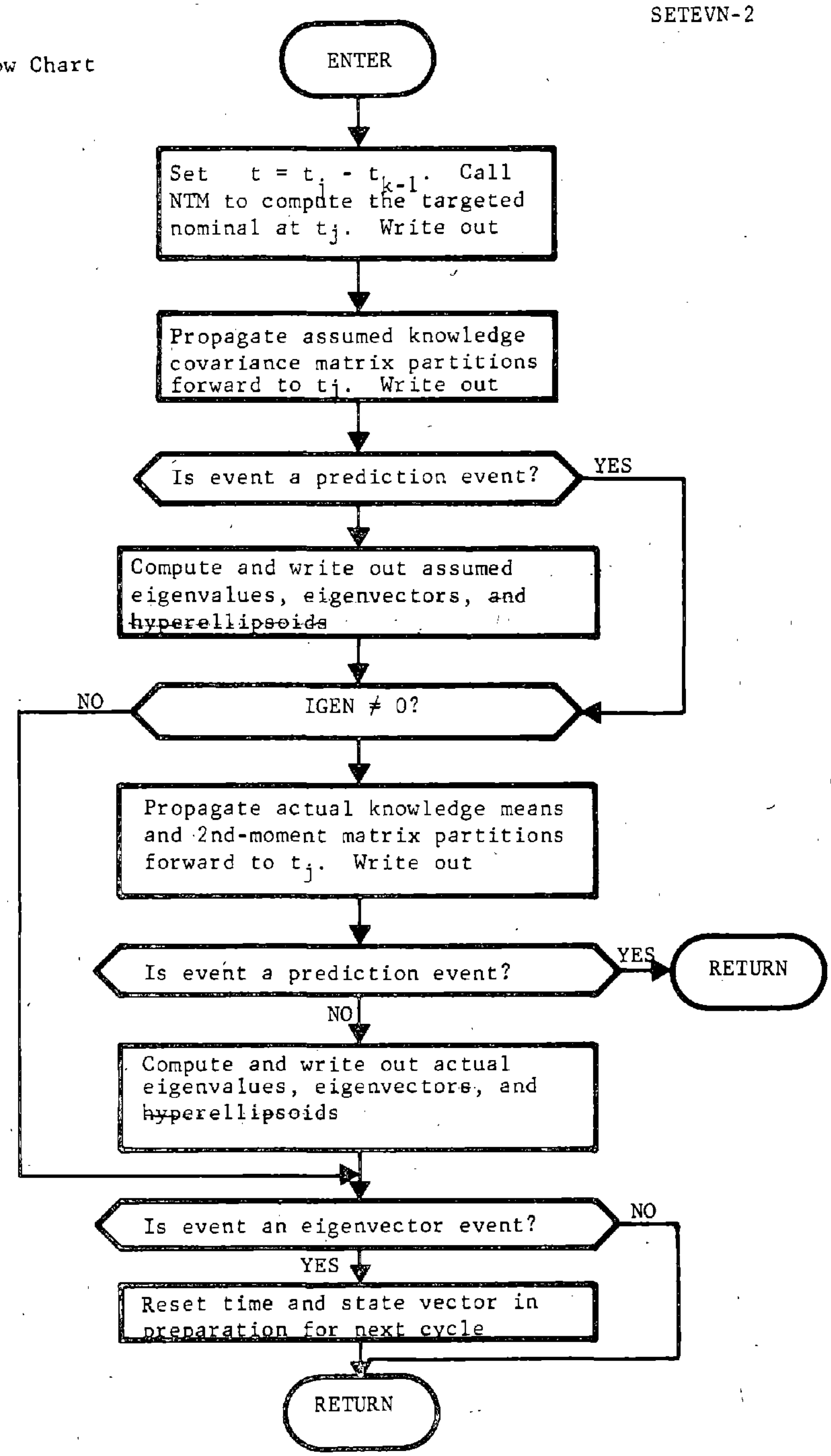
COMMON USED:	CXSU	XCSV	XCU	XCV	CXXS	FOP	FOV
	IPRT	ISTMC	PS	P	Q	U0	V0
	XLAB	IGEN	GU	GV	GCXW	GCXSW	GP
	GCXXS	GCXU	GCXV	GPS	GCXSU	GCXSV	QPR
	RPR	EXT	EXST				

SETEVN Analysis

Before executing any event in the error analysis/generalized covariance analysis program subroutine SETEVN is called to perform a series of computations that are common to all events. Subroutine SETEVN computes the targeted nominal trajectory at t_j and propagates the assumed and actual knowledge covariance partitions at t_{k-1} -- the time of the previous event or measurement -- forward to time t_j using the propagation equations found in subroutine GNAVM. The actual estimation error means are also propagated forward to t_j using the propagation equations found in subroutine MEAN.

For any event other than a prediction event, subroutine SETEVN also computes eigenvalues, eigenvectors of the position and velocity partitions of the assumed and actual knowledge covariance at t_j.

SETEVN Flow Chart



SUBROUTINE SET1

PURPOSE: TO INITIALIZE FLAGS FOR USE BY INTEGRATION ROUTINES

CALLING SEQUENCE: CALL SET1(Y,DJ)

ARGUMENTS:

Y	I	INITIAL 6 ELEMENT STATE VECTOR
DJ	I	JULIAN DATE CORRESPONDING TO STATE VECTOR Y

SUBROUTINES REQUIRED:

CALJUL
DSHIFT
DZERO

COMMON USED:

NBM	RETRO
NBOPTH	
SMU	
IBURN	

COMMON COMPUTED:

YMDIC	XDD	ICENT
HMSIC	XV	GM
T	XVD	IND
X	XVDD	H
XD	NB	ISUN
ITERS	NEQ	NOFC
NCONDT	NOCOWL	NTSEQS
NSECTN	NSTR	CETOL
SECTIM		

```
SUBROUTINE SHIFT

PURPOSE:  TO SHIFT A DOUBLE PRECISION ARRAY TO ANOTHER LOCATION

CALLING SEQUENCE:  CALL SHIFT(AIN,K,BOUT)

ARGUMENTS:  AIN   ARRAY TO BE SHIFTED
            K     NUMBER OF CONTIGUOUS VALUES TO BE SHIFTED
            BOUT  RECEIVING ARRAY
```

```
SUBROUTINE SL0020

PURPOSE:  TO MINIMIZE F(X)

CALLING SEQUENCE: CALL SL0020(X,Y,TS)

ARGUMENTS:
          X       I 3 VECTOR OF VALUES OF X
          Y       I 3 VECTOR OF F(X) VALUES CORRESPONDING TO VECTOR X
          TS      O PROJECTED VALUE OF X FOR WHICH F(X) IS MINIMUM

SUBROUTINES REQUIRED:
          NONE
```

```
SUBROUTINE   STAPRL

PURPOSE:  TO COMPUTE THE PARTIAL DERIVATIVES OF STATION LOCATION
          ERRORS.

CALLING SEQUENCE:  CALL STAPRL(AL,ALON,ALAT,PAT2,VEC,PA)

ARGUMENT:    AL        I   ALTITUDE OF THE STATION

             ALAT      I   LATITUDE OF THE STATION

             ALON      I   LONGITUDE OF THE STATION

             PA        O   PARTIAL OF STATION POSITION AND VELOCITY
                           WITH RESPECT TO ALTITUDE, LATITUDE AND
                           LONGITUDE

             PAT2      I   LONGITUDE + OMEGA*(CURRENT TIME-LAUNCH
                           TIME)

             VEC           UNUSED

SUBROUTINES SUPPORTED: TRAKS   TRAKM

LOCAL SYMBOLS:  G1        SINE OF LATITUDE

                G2        COSINE OF LATITUDE

                G3        SINE(PHI +OMEGA(T-UNIVT))

                G4        COSINE(PHI +OMEGA(T-UNIVT))
                          WHERE  PHI   =LONGITUDE
                                 OMEGA=EARTH ROTATION RATE
                                 T     =TIME
                                 UNIVT=UNIVERSAL TIME

                G5        SINE OF OBLIQUITY OF EARTH

                G6        COSINE OF OBLIQUITY OF EARTH

                OMEG      OMEGA IN PROPER UNITS

COMMON USED:              EPS    OMEGA   TM
```

STAPRL Analysis

The ecliptic components of the position and velocity of a tracking station relative to the Earth are related to station location parameters R, θ, and φ through the following set of equations:

$$X_s = R \cos\theta \cos G$$

$$Y_s = R \cos\theta \cos\epsilon \sin G + R \sin\theta \sin\epsilon$$

$$Z_s = -R \cos\theta \sin\epsilon \sin G + R \sin\theta \cos\epsilon$$

$$\dot{X}_s = -\omega R \cos\theta \sin G$$

$$\dot{Y}_s = \omega R \cos\theta \cos\epsilon \cos G$$

$$\dot{Z}_s = -\omega R \cos\theta \sin\epsilon \cos G$$

where $G = \phi + \omega(t - T)$, and T is the universal time at some epoch(usually launch time).

Subroutine STAPRL computes the negative of the partials of the previous quantities with respect to the station location parameters R, θ, and φ. These partials are summarized below:

$$-\frac{\partial X_s}{\partial R} = -\cos\theta \cos G$$

$$-\frac{\partial X_s}{\partial \theta} = R \sin\theta \cos G$$

$$-\frac{\partial X_s}{\partial \phi} = R \cos\theta \sin G$$

$$-\frac{\partial Y_s}{\partial R} = -\left[\sin\epsilon \sin\theta + \cos\epsilon \cos\theta \sin G\right]$$

$$-\frac{\partial Y_s}{\partial \theta} = R \cos\epsilon \sin\theta \sin G - R \sin\epsilon \cos\theta$$

$$-\frac{\partial Y_s}{\partial \phi} = -R \cos\epsilon \cos\theta \cos G$$

$$-\frac{\partial Z_s}{\partial R} = \sin\epsilon \cos\theta \sin G - \cos\epsilon \sin\theta$$

$$-\frac{\partial Z_s}{\partial \theta} = -\left[R \sin \epsilon \sin \theta \sin G + R \cos \epsilon \cos \theta\right]$$

$$-\frac{\partial Z_s}{\partial \phi} = R \sin \epsilon \cos \theta \cos G$$

$$-\frac{\partial \dot{X}_s}{\partial R} = \omega \cos \theta \sin G$$

$$-\frac{\partial \dot{X}_s}{\partial \theta} = -\omega R \sin \theta \sin G$$

$$-\frac{\partial \dot{X}_s}{\partial \phi} = \omega R \cos \theta \cos G$$

$$-\frac{\partial \dot{Y}_s}{\partial R} = -\omega \cos \theta \cos \epsilon \cos G$$

$$-\frac{\partial \dot{Y}_s}{\partial \theta} = \omega R \cos \epsilon \sin \theta \cos G$$

$$-\frac{\partial \dot{Y}_s}{\partial \phi} = \omega R \cos \epsilon \cos \theta \sin G$$

$$-\frac{\partial \dot{Z}_s}{\partial R} = \omega \sin \epsilon \cos \theta \cos G$$

$$-\frac{\partial \dot{Z}_s}{\partial \theta} = -\omega R \sin \epsilon \sin \theta \cos G$$

$$-\frac{\partial \dot{Z}_s}{\partial \phi} = -\omega R \sin \epsilon \cos \theta \sin G$$

MAIN PROGRAM STEAPE

PURPOSE: TO CONTROL THE ERROR ANALYSIS MODE OF STEAP

CALLING SEQUENCE: NONE

ARGUMENTS: NONE

SUBROUTINES REQUIRED: DATA ERRAN

SUBROUTINE STMPR

PURPOSE: TO PRINT OUT THE TRANSPOSES OF THE STATE TRANSITION MATRIX PARTITIONS PHI, TXXS, TXW, AND TXU OVER AN ARBITRARY INTERVAL OF TIME.

CALLING SEQUENCE: CALL STMPR(TRTM1,TRTM2)

ARGUMENT:	TRTM1	I	TIME AT BEGINNING OF INTERVAL OVER WHICH STATE TRANSITION MATRIX PARTITIONS HAVE BEEN COMPUTED
	TRTM2	I	TIME AT END OF INTERVAL OVER WHICH STATE TRANSITION MATRIX PARTITIONS HAVE BEEN COMPUTED

SUBROUTINES SUPPORTED:	PRINT4	SETEVS	GUISIM	GUISS	PRESIM
	PRINT3	SETEVN	GUIDM	GUID	PRED
	PROBE	PROBES			

COMMON USED:	NDIM1	NDIM2	PHI	TXU	TXXS
	XLAB	XSL	XU		
	NDIM4	TXW			

SUBROUTINE STVCPR

PURPOSE: PRINT THE EPOCH, T, AND PRINT THE STATE VECTOR XI IN SEVERAL COORDINATE SYSTEMS (GEOCENTRIC ECLIPTIC, HELIOCENTRIC ECLIPTIC, ROTATING BARYCENTRIC, ROTATING L1-CENTRIC, AND ROTATING L2-CENTRIC)

CALLING SEQUENCE: CALL STVCPR(XI,T)

ARGUMENTS:

XI	STATE VECTOR
T	EPOCH OF STATE VECTOR

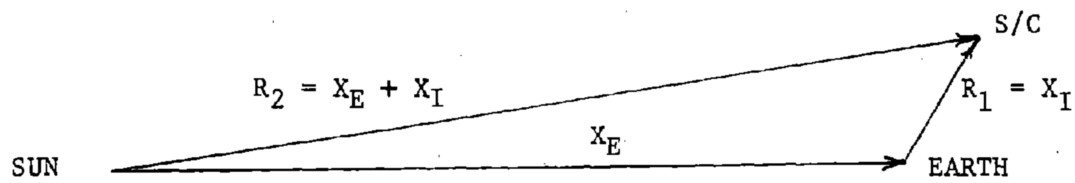
LOCAL VARIABLES:

AUX	INTERMEDIATE VECTOR
EL	ROTATION VECTOR
R	VECTOR TO BE PRINTED
RB	SUN-TO-BARYCENTER DISTANCE
RM	RADIUS MAGNITUDE TO BE PRINTED
RX	INTERMEDIATE VALUE
THETA	ROTATION MATRIX
VM	VELOCITY MAGNITUDE TO BE PRINTED
VP	VELOCITY DUE TO ROTATING FRAME OF REFERENCE
XB	HELIOCENTRIC STATE VECTOR OF BARYCENTER
XM	HELIOCENTRIC STATE VECTOR OF MOON
XP	HELIOCENTRIC STATE VECTOR OF EARTH

COMMON USED: DATEJ MUPLAN TRTMB

STVCPR Analysis

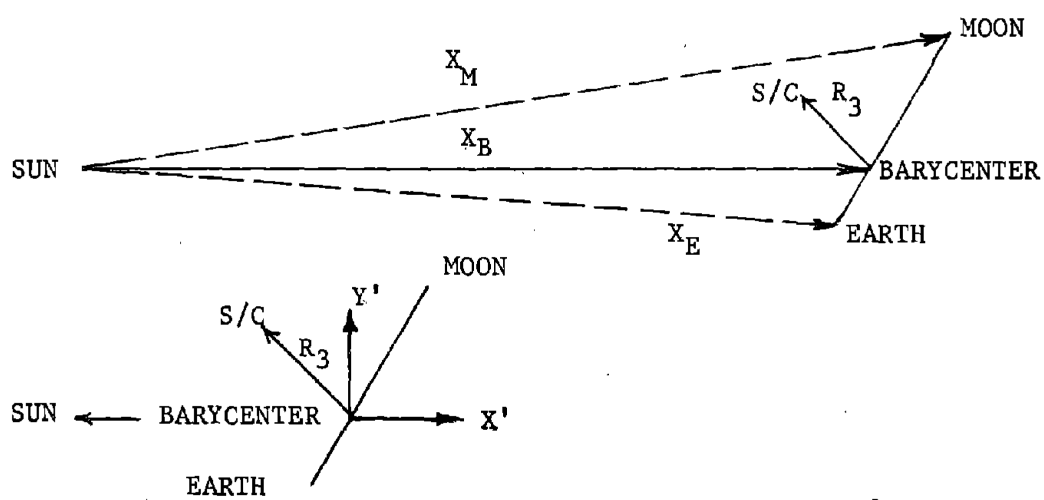
Subroutine STVCPR prints the state vector XI at epoch T in several coordinate systems: geocentric ecliptic, heliocentric ecliptic, rotating barycentric, rotating L1-centric and rotating L2-centric. The state vector XI is read from the GTDS (Cowell) file in the geocentric ecliptic coordinate system, and so the first print involves no transformation. The earth's heliocentric ecliptic state is then added to XI for the second print, XI in heliocentric ecliptic state.



Next the moon's heliocentric ecliptic state is obtained, and the barycenter's state is computed

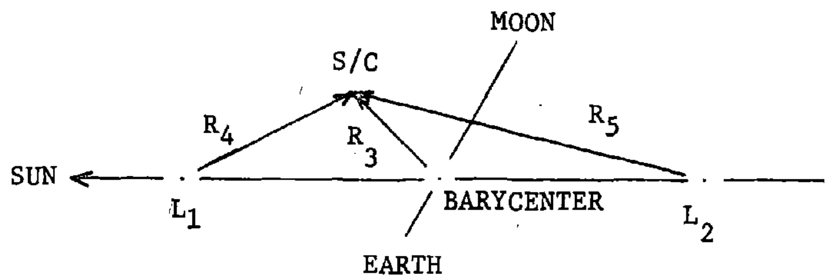
$$\vec{X}_B = \frac{1}{\mu_E + \mu_M} (\mu_E \vec{X}_E + \mu_M \vec{X}_M)$$

The rotating barycentric frame has the same Z-axis as the ecliptic, but rotates so that the X'-axis is always the barycenter-sun line. A matrix is defined to effect the rotation of the position and velocity components.



Then the velocity change due to the rotation of the frame and to the spacecraft position in the frame is added to the velocity, and the third print is the rotating barycentric representation of XI.

For the last two representations, the origin of the coordinate system is simply translated back and forth along the X'-axis to the Lagrangian L1 and L2 points respectively.



SUBROUTINE SYMTRK

PURPOSE: SYMMETRIZE A SQUARE MATRIX

CALLING SEQUENCE: CALL SYMTRK(ARRAY,M)

ARGUMENTS: ARRAY NAME OF THE M-BY-M MATRIX
M DIMENSION OF THE SQUARE ARRAY

SUBROUTINES REQUIRED: NONE

COMMON USED: HALF

SUBROUTINE SYMTRZ

PURPOSE: FILL IN THE UPPER-RIGHT TRIANGLE OF A SYMMETRIC SQUARE MATRIX WHOSE LOWER-LEFT TRIANGLE WAS INPUT

CALLING SEQUENCE: CALL SYMTRZ(P,K,N)

ARGUMENTS: P SQUARE MATRIX TO BE COMPLETED (N-BY-N)
K NUMBER OF ROWS IN USE
N ACTUAL DIMENSION OF MATRIX

SUBROUTINES REQUIRED: NONE

SUBROUTINE TIME

PURPOSE: CONVERT A TIME IN SECONDS INTO DAYS, HOURS, MINUTES, SECONDS

CALLING SEQUENCE: CALL TIME(TSEC,JDAY,JHR,JMIN,XSEC)

ARGUMENTS: TSEC TIME IN SECONDS
JDAY NUMBER OF DAYS
JHR NUMBER OF HOURS
JMIN NUMBER OF MINUTES
XSEC NUMBER OF SECONDS

SUBROUTINE TINE

PURPOSE: TO COMPUTE THE JULIAN DATE, EPOCH 1900, FROM THE CALENDAR DATE OR TO COMPUTE THE CALENDAR DATE FROM THE JULIAN DATE.

CALLING SEQUENCE: CALL TINE(DAY,IYR,MO,IDAY,IHR,MIN,SEC,ICODE)

ARGUMENTS:

DAY	I/O	JULIAN DATE, EPOCH 1900
IYR	O/I	CALENDAR YEAR
MO	O/I	CALENDAR MONTH
IDAY	O/I	CALENDAR DAY
IHR	O/I	HOUR OF THE DAY
MIN	O/I	MINUTE OF HOUR
SEC	O/I	FRACTIONAL SECONDS
ICODE	I	OPERATIONAL MODE = 1, INDICATES THE JULIAN DATE IS INPUT, CALENDAR DATE IS OUTPUT = 0, INDICATES THE CALENDAR DATE IS INPUT, JULIAN DATE IS OUTPUT

SUBROUTINES SUPPORTED: DATA

SUBROUTINES REQUIRED: NONE

LOCAL SYMBOLS:

IA	NUMBER OF CENTURIES
IB	YEARS IN PRESENT CENTURY
IP	NUMBER OF MONTH (BASED ON MARCH AS NUMBER ZERO)
IQ	NUMBER OF YEARS
IR	NUMBER OF CENTURIES DIVIDED BY 4
IS	NUMBER OF YEARS SINCE LAST 400 YEAR SECTION BEGAN
IT	NUMBER OF LEAP YEARS IN PRESENT CENTURY
IU	NUMEER OF YEARS SINCE LAST LEAP YEAR
IV	NUMBER OF DAYS IN LAST YEAR

IX	INTERMEDIATE INTEGER
J	INTERMEDIATE INTEGER
JD	NUMBER OF DAYS IN JULIAN DATE
P	JULIAN DATE
R	FRACTIONAL PORTION OF DAY IN JULIAN DATE

SUBROUTINE TRAKM

PURPOSE: COMPUTE THE OBSERVATION MATRIX AND ITS AUGMENTATIONS

CALLING SEQUENCE: CALL TRAKM(RV,ITRK,NR)

ARGUMENTS:
- RV — STATE VECTOR AT TIME OF MEASUREMENT
- ITRK — TYPE OF MEASUREMENT
- NR — NUMBER OF ROWS IN OBSERVATION MATRIX (I.E., THE DIMENSION OF THE MEASUREMENT)

LOCAL SYMBOLS:

AL	RADIUS OF STATION IA
ALAT	LATITUDE OF STATION IA
ALON	LONGITUDE OF STATION IA
CE	COSINE OF EARTH OBLIQUITY
COAL	COSINE OF STAR-PLANET ANGLE
CP	COSINE OF PAT2
DEL	TIME IN DAYS FROM INITIAL TIME ON FILE
G2	Y-COMPONENT, EQUATORIAL LOCATION, STATION IA
G3	Z-COMPONENT, EQUATORIAL LOCATION, STATION IA
G5	YDOT-COMPONENT, EQUATORIAL LOCATION, STATION IA
GELS	GEOCENTRIC EQUATORIAL LOCATION OF STATION IA
IA	INDEX OF STATION MAKING MEASUREMENT
IC	COLUMN NUMBER IN MATRIX AL
ICD	INDEX TO LOCATE AUGMENTATION PARAMETERS FOR IA
IEND	LAST AUGMENTATION PARAMETER TO BE CHECKED
IK	COLUMN NUMBER IN MATRIX AM
IL	COLUMN NUMBER IN MATRIX AN
IR	REFERENCE INDEX, STAR-PLANET ANGLES
NA	STAR-PLANET ANGLE CURRENTLY BEING COMPUTED
NC	LAST STAR-PLANET ANGLE TO BE COMPUTED
PA	ARRAY OF PARTIALS D(RF)/D(GELS)
PAR	ARRAY OF PARTIALS D(RF)/D(R-PLANET)
PAT1	INTERMEDIATE VARIABLE
PAT2	HOUR ANGLE OF STATION IA
RADNEP	RADIUS OF EPHEMERIS PLANET (EARTH)
RF	STATE VECTOR WRT EARTH OR WRT STATION IA
RFMAG	MAGNITUDE OF RF
RRATE	RANGE-RATE
RS	SPIN RADIUS OF STATION IA
SE	SINE OF EARTH OBLIQUITY
SIAL	SINE OF STAR-PLANET ANGLE
SP	SINE OF PAT2
ZTAR	UNIT VECTOR, DIRECTION OF STAR NA

```
COMMON COMPUTED:  AL     AM     AN     G      H
                  (AL IN COMMON BLOCK STM IS CALLED AAL IN TRAKM)

COMMON USED:   ALNGTH  DELTM   EPS     IAUGIN  OMEGA   RAD     RADIUS
               SAL     SLAT    SLON    TM      TRTMB   TRTM1   UNIVTM
               UST     VST     WST
```

TRAKM Analysis

Subroutine TRAKM computes all observation matrix partitions for the measurement type indicated by ITRK. The number of rows, NR, in the measurement and the observation matrix partitions is also computed.

The linearized observation equation can be written as

$$y = Hx + Mx_s + Gu + Lv + Nw$$

where y is the observable, x is the spacecraft state, and x_s, u, v, and w are solve-for, dynamic consider, measurement consider, and ignore parameter vectors, respectively. The function of subroutine TRAKM is to compute the observation matrix partitions H, M, G, L, and N, which indicate the sensitivity of the observable v to changes in x, x_s, u, v, and w, respectively, in the error analysis/generalized covariance analysis program. The matrix N is computed only for a generalized covariance analysis.

In the remainder of this section the measurement equation and all partial derivatives required to construct the H, M, G, and L observation matrix partitions will be summarized for each measurement type.

A. Range Measurement

A range measurement has form

$$\rho = \rho(\bar{X}, R, \theta, \phi, t)$$

where R, θ, and ∅ are the radius, latitude, and longitude of the relevant tracking station.

More explicitly,

$$\rho = \left[(X - X_E - X_S)^2 + (Y - Y_E - Y_S)^2 + (Z - Z_E - Z_S)^2 \right]^{\frac{1}{2}}$$

where X, Y, Z = inertial position components of spacecraft
X_E, Y_E, Z_E = inertial position components of Earth
X_S, Y_S, Z_S = station position components relative to Earth.

X_S, Y_S, and Z_S are related to R, θ, and ∅ as follows:

$$X_S = R \cos\theta \cos G$$

$$Y_S = R \cos\theta \cos\epsilon \sin G + R \sin\theta \sin\epsilon$$

$$Z_S = -R \cos\theta \sin\epsilon \sin G + R \sin\theta \cos\epsilon$$

where ϵ is the obliquity of the Earth, and

$$G = \emptyset + GHA$$

where GHA is the Greenwich hour angle at time t .

Partials of ρ with respect to spacecraft state are given by

$$\frac{\partial \rho}{\partial X} = \frac{1}{\rho}(X - X_E - X_S) \qquad \frac{\partial \rho}{\partial \dot{X}} = 0$$

$$\frac{\partial \rho}{\partial Y} = \frac{1}{\rho}(Y - Y_E - Y_S) \qquad \frac{\partial \rho}{\partial \dot{Y}} = 0$$

$$\frac{\partial \rho}{\partial Z} = \frac{1}{\rho}(Z - Z_E - Z_S) \qquad \frac{\partial \rho}{\partial \dot{Z}} = 0$$

Partials of ρ with respect to R, θ, and ∅ are given by

$$\frac{\partial \rho}{\partial R} = \frac{\partial \rho}{\partial X_S} \cdot \frac{\partial X_S}{\partial R} + \frac{\partial \rho}{\partial Y_S} \cdot \frac{\partial Y_S}{\partial R} + \frac{\partial \rho}{\partial Z_S} \cdot \frac{\partial Z_S}{\partial R}$$

$$\frac{\partial \rho}{\partial \theta} = \frac{\partial \rho}{\partial X_S} \cdot \frac{\partial X_S}{\partial \theta} + \frac{\partial \rho}{\partial Y_S} \cdot \frac{\partial Y_S}{\partial \theta} + \frac{\partial \rho}{\partial Z_S} \cdot \frac{\partial Z_S}{\partial \theta}$$

$$\frac{\partial \rho}{\partial \emptyset} = \frac{\partial \rho}{\partial X_S} \cdot \frac{\partial X_S}{\partial \emptyset} + \frac{\partial \rho}{\partial Y_S} \cdot \frac{\partial Y_S}{\partial \emptyset} + \frac{\partial \rho}{\partial Z_S} \cdot \frac{\partial Z_S}{\partial \emptyset}$$

where

$$\frac{\partial \rho}{\partial X_S} = -\frac{\partial \rho}{\partial X}, \quad \frac{\partial \rho}{\partial Y_S} = -\frac{\partial \rho}{\partial Y}, \quad \frac{\partial \rho}{\partial Z_S} = -\frac{\partial \rho}{\partial Z}$$

and the negatives of the partials of X_S, Y_S, and Z_S with respect to R, θ, and $\emptyset$ are summarized in the subroutine STAPRL analysis.

B. Range-rate measurement $\dot{\rho}$.

A range-rate measurement has form

$$\dot{\rho} = \dot{\rho}(\vec{X}, R, \theta, \emptyset, t)$$

where all arguments have been defined previously. More explicitly,

$$\dot{\rho} = \frac{\rho_x \dot{\rho}_x + \rho_y \dot{\rho}_y + \rho_z \dot{\rho}_z}{\rho}$$

where

$$\begin{aligned} \rho_x &= X - X_E - X_S & \dot{\rho}_x &= \dot{X} - \dot{X}_E - \dot{X}_S \\ \rho_y &= Y - Y_E - Y_S & \dot{\rho}_y &= \dot{Y} - \dot{Y}_E - \dot{Y}_S \\ \rho_z &= Z - Z_E - Z_S & \dot{\rho}_z &= \dot{Z} - \dot{Z}_E - \dot{Z}_S \end{aligned}$$

$\dot{X}_S$, $\dot{Y}_S$, and $\dot{Z}_S$ are related to R, θ, and $\emptyset$ as follows:

$$\begin{aligned} \dot{X}_S &= -\omega R \cos\theta \sin G \\ \dot{Y}_S &= \omega R \cos\theta \cos\epsilon \cos G \\ \dot{Z}_S &= -\omega R \cos\theta \sin\epsilon \cos G \end{aligned}$$

where ω is the rotational rate of the Earth.

Partials of $\dot{\rho}$ with respect to spacecraft state are given by

$$\frac{\partial \dot{\rho}}{\partial X} = \frac{\dot{\rho}_x}{\rho} - \frac{\rho_x \dot{\rho}}{\rho^2} \qquad \frac{\partial \dot{\rho}}{\partial \dot{X}} = \frac{\rho_x}{\rho}$$

$$\frac{\partial \dot{\rho}}{\partial Y} = \frac{\dot{\rho}_y}{\rho} - \frac{\rho_y \dot{\rho}}{\rho^2} \qquad \frac{\partial \dot{\rho}}{\partial \dot{Y}} = \frac{\rho_y}{\rho}$$

$$\frac{\partial \dot{\rho}}{\partial Z} = \frac{\dot{\rho}_z}{\rho} - \frac{\rho_z \dot{\rho}}{\rho^2} \qquad \frac{\partial \dot{\rho}}{\partial \dot{Z}} = \frac{\rho_z}{\rho}$$

The partial of $\dot{\rho}$ with respect to R is given by

$$\frac{\partial \dot{\rho}}{\partial R} = \frac{\partial \dot{\rho}}{\partial X_S} \cdot \frac{\partial X_S}{\partial R} + \frac{\partial \dot{\rho}}{\partial Y_S} \cdot \frac{\partial Y_S}{\partial R} + \frac{\partial \dot{\rho}}{\partial Z_S} \cdot \frac{\partial Z_S}{\partial R} +$$

$$\frac{\partial \dot{\rho}}{\partial \dot{X}_S} \cdot \frac{\partial \dot{X}_S}{\partial R} + \frac{\partial \dot{\rho}}{\partial \dot{Y}_S} \cdot \frac{\partial \dot{Y}_S}{\partial R} + \frac{\partial \dot{\rho}}{\partial \dot{Z}_S} \cdot \frac{\partial \dot{Z}_S}{\partial R}$$

where

$$\frac{\partial \dot{\rho}}{\partial X_S} = - \frac{\partial \dot{\rho}}{\partial X} \text{ , etc.}$$

and $$\frac{\partial \dot{\rho}}{\partial \dot{X}_S} = - \frac{\partial \dot{\rho}}{\partial \dot{X}} \text{ , etc.}$$

The negatives of the partials of X_S, Y_S, Z_S, $\dot{X}_S$, $\dot{Y}_S$, and $\dot{Z}_S$ with respect to R, θ, and ϕ are summarized in the subroutine STAPRL analysis. Partials of $\dot{\rho}$ with respect to θ and ϕ are treated similarly.

C. Star-planet angle measurement α.

A star-planet angle measurement has form α

$$\alpha = \alpha\ (\vec{X}, a, e, i, \Omega, \omega, M)$$

where a, e, i, Ω, ω, and M are the standard set of target planet orbital elements.

If we differ $\vec{\rho} = (\rho_x, \rho_y, \rho_z)$ to be the position of the target planet relative to the spacecraft and (u, v, w) to be the direction cosines of the relevant star, then

$$\alpha = \cos^{-1} \left[\frac{1}{\rho} (u\rho_x + v\rho_y + w\rho_z) \right]$$

where

$$\rho_x = X_p - X, \quad \rho_y = Y_p - Y, \quad \rho_z = Z_p - Z,$$

and (X_p, Y_p, Z_p) represent the position coordinates of the target planet.

Partials of α with respect to spacecraft state are given by

$$\frac{\partial \alpha}{\partial X} = \frac{1}{\sin \alpha}\left(\frac{u}{\rho} - \frac{\rho_x \cos \alpha}{\rho^2}\right) \qquad \frac{\partial \alpha}{\partial \dot{X}} = 0$$

$$\frac{\partial \alpha}{\partial Y} = \frac{1}{\sin \alpha}\left(\frac{v}{\rho} - \frac{\rho_y \cos \alpha}{\rho^2}\right) \qquad \frac{\partial \alpha}{\partial \dot{Y}} = 0$$

$$\frac{\partial \alpha}{\partial Z} = \frac{1}{\sin \alpha}\left(\frac{w}{\rho} - \frac{\rho_z \cos \alpha}{\rho^2}\right) \qquad \frac{\partial \alpha}{\partial \dot{Z}} = 0$$

where

$$\sin \alpha = + \left[1 - \cos^2 \alpha\right]^{\frac{1}{2}} .$$

D. Apparent planet diameter measurement β .

An apparent planet diameter measurement has form

$$\beta = \beta(\vec{X}, a, e, i, \Omega, \omega, M)$$

where all arguments have been defined previously.

Defining $\vec{\rho} = (\rho_x, \rho_y, \rho_z)$ to be the position of the target planet relative to the spacecraft and R_p to be the radius of the target planet, the apparent planet diameter can then be written as

$$\beta = 2 \sin^{-1}\left(\frac{R_p}{\rho}\right)$$

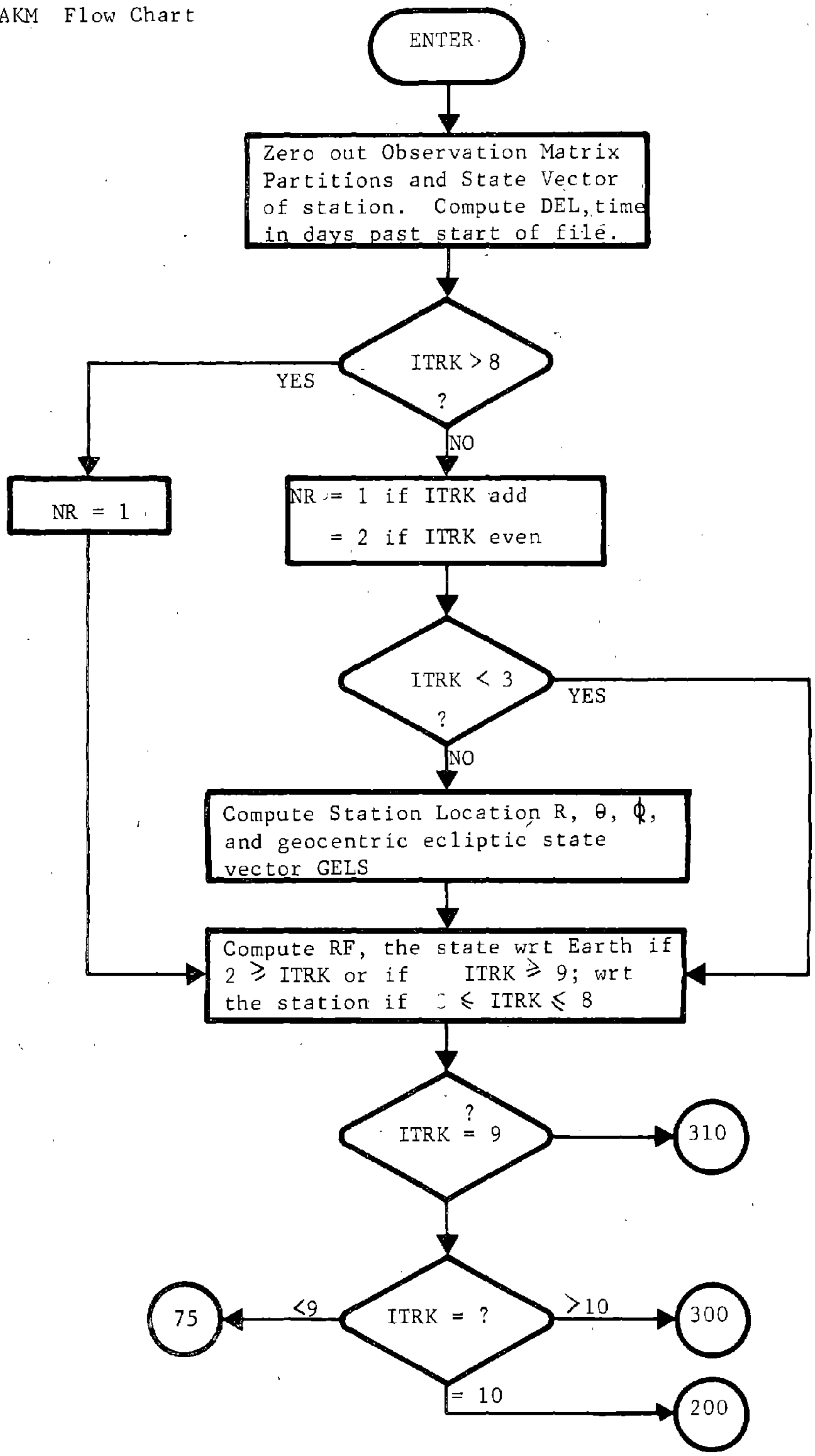
Partials of β with respect to spacecraft state are given by

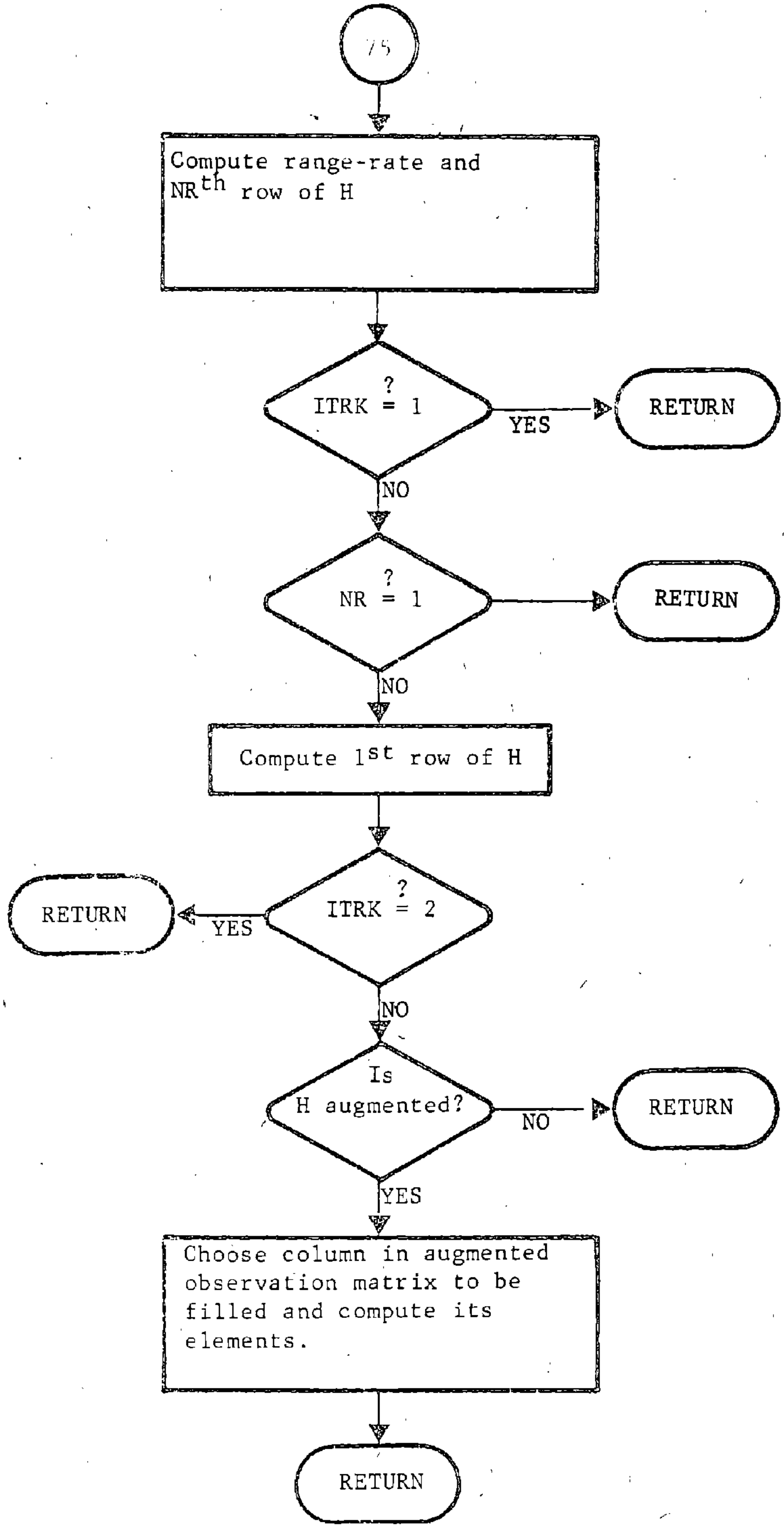
$$\frac{\partial \beta}{\partial X} = \frac{2 R_p \rho_x}{\rho^2 \left[\rho^2 - R_p^{\,2}\right]^{\frac{1}{2}}} \qquad \frac{\partial \beta}{\partial \dot{X}} = 0$$

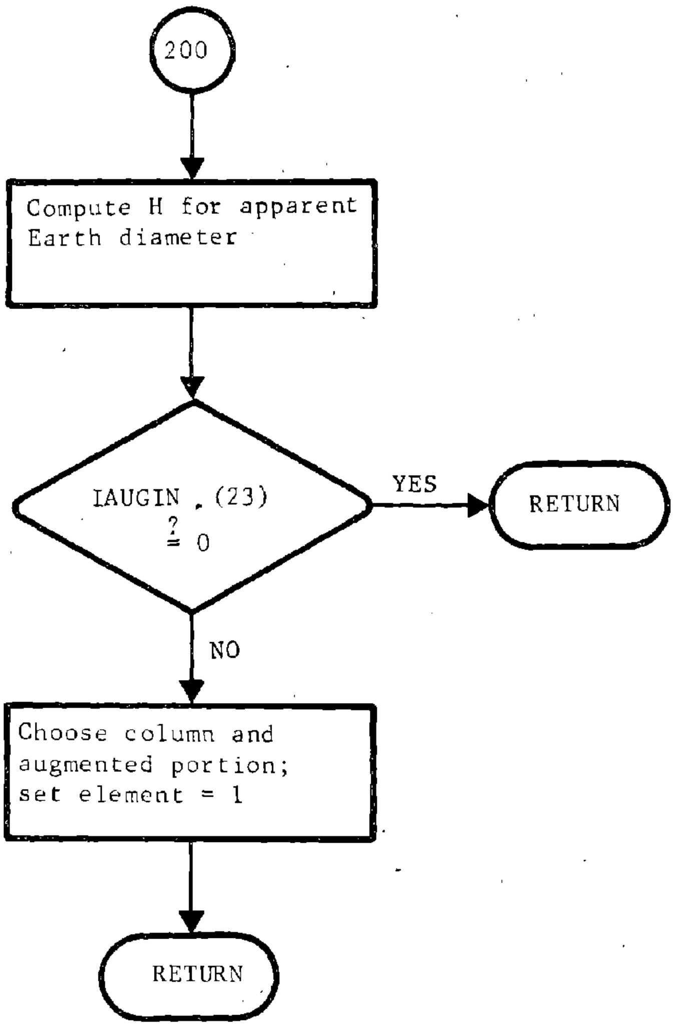
$$\frac{\partial \beta}{\partial Y} = \frac{2 R_p \rho_y}{\rho^2 \left[\rho^2 - R_p^{\,2}\right]^{\frac{1}{2}}} \qquad \frac{\partial \beta}{\partial \dot{Y}} = 0$$

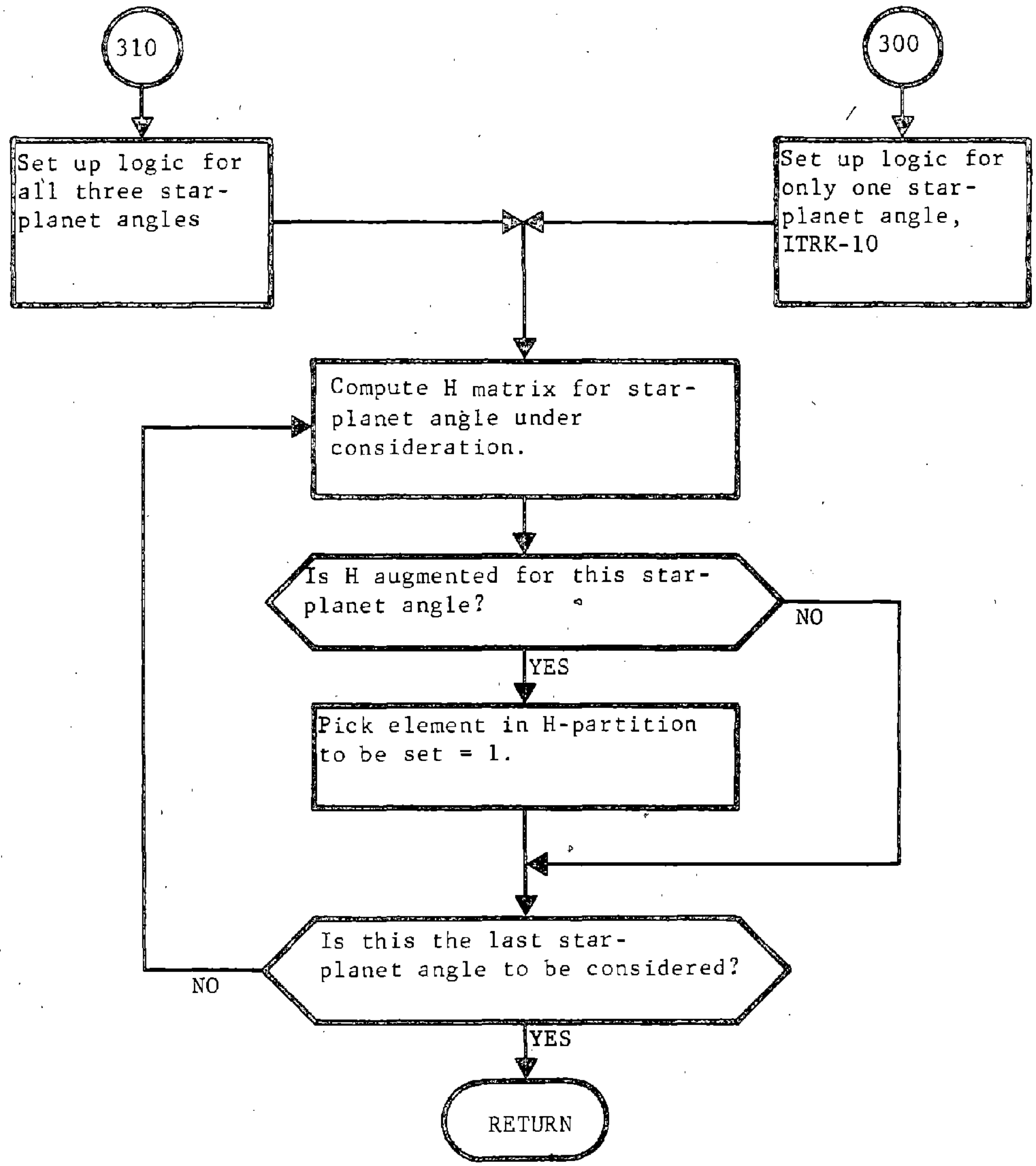
$$\frac{\partial \beta}{\partial Z} = \frac{2 R_p \rho_z}{\rho^2 \left[\rho^2 - R_p^{\,2}\right]^{\frac{1}{2}}} \qquad \frac{\partial \beta}{\partial \dot{Z}} = 0$$

TRAKM Flow Chart




75
Compute range-rate and NR^{th} row of H
ITRK ?= 1
YES
RETURN
NO
NR ?= 1
RETURN
NO
Compute 1^{st} row of H
ITRK ?= 2
YES
RETURN
NO
Is H augmented?
NO
RETURN
YES
Choose column in augmented observation matrix to be filled and compute its elements.
RETURN


200
Compute H for apparent Earth diameter
IAUGIN (23) ?= 0
YES
RETURN
NO
Choose column and augmented portion; set element = 1
RETURN


310
300
Set up logic for all three star-planet angles
Set up logic for only one star-planet angle, ITRK-10
Compute H matrix for star-planet angle under consideration.
Is H augmented for this star-planet angle?
NO
YES
Pick element in H-partition to be set = 1.
Is this the last star-planet angle to be considered?
NO
YES
RETURN

SUBROUTINE TRJTRY (PRELIM ENTRY POINT)

PURPOSE: DUMMY LINK WITH NON HALO ORBIT OPTIONS

CALLING SEQUENCE: CALL TRJTRY

SUBROUTINE TRNSPS

PURPOSE: TO FORM THE TRANSPOSE OF A MATRIX
CALLING SEQUENCE: CALL TRNSPS (A,B,M,N)

ARGUMENTS:

A	I	MATRIX TO BE TRANSPOSED
B	O	RECEIVING MATRIX
M	I	NUMBER OF ROWS IN A AND COLUMNS IN B
N	I	NUMBER OF COLUMNS IN A AND ROWS IN B

SUBROUTINES REQUIRED:
NONE

SUBROUTINE ZERMAT

PURPOSE: TO ZERO OUT A MATRIX

CALLING SEQUENCE: CALL ZERMAT(ARR,NR,NC)

ARGUMENTS:

ARR	ARRAY TO BE ZEROED OUT
NR	NUMBER OF ROWS
NC	NUMBER OF COLUMNS

5. REFERENCES

[1] Lee, B. G., et al: Interplanetary Trajectory Error Analysis, Final Report for Contract NAS8-21120, MMC Report MCR-67-441, December 1967.

[2] Lee, B. G., et al: Space Trajectories Error Analyses Programs, Final Report for Contract NAS1-8745, NASA CR-66818, August 1969.

[3] Vogt, E. D., et al: Space Trajectories Error Analyses Programs Version II, Final Report for Contracts NAS5-11795 and -11873, MCR-71-4, December 1971.

[4] Hong, P., et al: Low Thrust Orbit Determination Program, Final Report for Contract NAS1-11686, NASA CR-112256, December 1972.

[5] Hong, P., et al: Mission Analysis Program for Solar Electric Propulsion, Final Report for Contract NAS8-29666, MMC Report MCR-74-82, March 1974.

[6] Lee, G. and Boain, R.: Propellant Requirements for Midcourse Velocity Corrections, Journal of Spacecraft and Rockets, Vol. 10, No. 12, December 1973.

[7] Hoffman, L. and Young, G.: Approximation to the Statistics of Midcourse Velocity Corrections, NASA TN D-5381.

[8] Pu, C. L. and Edelbaum, T. N.: Four-Body Trajectory Optimization, Final Report on NGR 22-009-207, Draper Laboratory, Inc., R-778, December 1973.

[9] Farquhar, R. W.: Future Missions for Libration-Point Satellites, Astronautics and Aeronautics, May 1969.

[10] Battin, R. H.: Astronautical Guidance, McGraw-Hill Publishing Co., 1964.

[11] Lancaster, E. R. and Blanchard, R. C.: A Unified Form of Lambert's Theirem, NASA Technical Note D-5368.